

**SISTEM PENYUSUNAN JADWAL PELAJARAN SEKOLAH
BERBASIS WEB DI
SMK NEGERI 1 PACITAN**

SKRIPSI

Diajukan kepada Fakultas Teknik
Universitas Negeri Yogyakarta
untuk Memenuhi Sebagian Persyaratan
guna Memperoleh Gelar Sarjana Pendidikan



Oleh :

Fajar Kurniawan

NIM 08520241021

**PROGRAM STUDI PENDIDIKAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
2012**

LEMBAR PERSETUJUAN

Skripsi yang berjudul **“SISTEM PENYUSUNAN JADWAL PELAJARAN
BERBASIS WEB DI SMKN 1 PACITAN”** yang disusun oleh Fajar Kurniawan,
NIM 08520241021 ini telah disetujui oleh pembimbing untuk diujikan.

Yogyakarta, // oktober 2012

Menyetujui,

Pembimbing






Totok Sukardiyono, M.T.

NIP. 19670903 199303 1 005

LEMBAR PENGESAHAN

Skripsi yang berjudul “**SISTEM PENYUSUNAN JADWAL PELAJARAN BERBASIS WEB DI SMKN 1 PACITAN**” yang disusun oleh Fajar Kurniawan, NIM 08520241021 ini TELAH DIPERTAHANKAN DI DEPAN Dewan Penguji pada tanggal 13 November 2012 dan dinyatakan lulus.

DEWAN PENGUJI

Nama	Jabatan	Tanda Tangan	Tanggal
Totok Sukardiyono, M.T.	Ketua penguji		18/12 2012
Handaru Jati, Ph.D.	Sekretaris		19/12 2012
Dr. Drs. Eko Marpanaji, M.T.	Penguji Utama		19/12 2012

Yogyakarta, Desember 2012

Fakultas Teknik UNY,

Dekan



Dr. Moch. Bruri Triyono

NIP. 19560216 198603 1 003

HALAMAN PERNYATAAN

Dengan ini saya menyatakan bahwa skripsi ini benar-benar karya saya sendiri. Sepanjang pengetahuan saya tidak terdapat karya atau pendapat yang ditulis atau diterbitkan orang lain kecuali sebagai acuan atau kutipan dengan mengikuti tata penulisan karya ilmiah yang telah lazim.

Tanda tangan dosen penguji yang tertera dalam halaman pengesahan adalah asli. Jika tidak asli, saya siap menerima sanksi ditunda yudisium pada periode berikutnya.

Yogyakarta, 11 oktober 2012

Yang menyatakan,



Fajar Kurniawan

NIM. 08520241021



MOTTO

Menunggu kesuksesan adalah tindakan sia-sia yang bodoh.

Jadikanlah kekecewaan masa lalu menjadi senjata sukses dimasa depan.

Kesulitan lebih cepat terlihat daripada kemudahan. Itu sebabnya mengeluh lebih cepat daripada bersyukur. ~ Mario Teguh

Hidup kadang membuat kita jatuh, tapi pasti kita punya pilihan untuk bangkit & berjuang lagi! ~ Andrey Wongso

Perjalanan Ribuan Mil Berawal dari Satu Langkah ~ Lao Tzu

PERSEMBAHAN

Dengan mengucapkan syukur Alhamdulillah, kupersembahkan karya kecilku ini untuk orang-orang yang kusayangi :

Ayah bunda tercinta, motivator terbesar dalam hidupku yang tak pernah jemu mendo'akan dan menyayangiku, atas semua pengorbanan dan kesabaran mengantarku sampai kini. Tak pernah cukup ku membalas cinta ayah bunda padaku.

Kakak-kakakku Yuliatin, Agung Wibowo

Sahabatku Ari Wibowo, Ridwan Danur yang selalu membantuku

Keluargku yang telah memberiku kelonggaran waktu sehingga aku dapat melaksanakan perkuliahan hingga penyusunan skripsi sampai tuntas

Sahabat-sahabatku seperjuangan di Kos Soropadan dan semua teman-teman yang tak mungkin penulis sebutkan satu-persatu, for u all I miss u forever

SISTEM PENYUSUNAN JADWAL PELAJARAN SEKOLAH BERBASIS WEB DI SMKN 1 PACITAN

Oleh
Fajar Kurniawan
NIM 08520241021

ABSTRAK

Penelitian ini bertujuan untuk mengembangkan Perangkat Lunak Sistem Penyusunan Jadwal Pelajaran Sekolah menggunakan bahasa pemrograman PHP dan database MySQL dan melakukan analisis kualitas pada aplikasi yang dikembangkan, khususnya pada faktor kualitas *User Interface*, *Functionality*, *Compatibility*, dan *Usability*.

Pengembangan Perangkat Lunak Sistem Penyusunan Jadwal Pelajaran Sekolah dilakukan dengan kaidah rekayasa perangkat lunak (*softwareengineering*) yaitu dimulai dari proses pendefinisian (*define*), perencanaan (*design*), pengembangan (*develop*), dan penyebaran (*disseminate*). Analisis faktor kualitas *User Interface*, *Functionality*, *Compatibility*, dan *Usability* dilakukan dengan metode kuesioner dengan responden guru dan staff kurikulum SMK Negeri 1 Pacitan.

Hasil pengembangan aplikasi yaitu Perangkat Lunak Sistem Penyusunan Jadwal Pelajaran Sekolah dalam bentuk *web application* yang digunakan melalui *web browser*. Hasil analisis kualitas menunjukkan bahwa aplikasi yang dikembangkan memenuhi semua standar faktor kualitas yang diujikan yaitu *User Interface*, *Functionality*, *Compatibility*, dan *Usability*.

Kata kunci : jadwal, perangkat lunak, (*Software Quality Assurance*), *web application*

KATA PENGANTAR

Segala puji dan syukur penulis panjatkan Kehadirat Tuhan Yang Maha Esa atas kuasa dan limpahan rahmat-Nya sehingga penulis dapat menyelesaikan penyusunan skripsi ini sebagai salah satu syarat untuk memperoleh gelar Sarjana Pendidikan di Fakultas Teknik Universitas Negeri Yogyakarta.

Penyusunan skripsi ini tidak terlepas dari bantuan, bimbingan, dan peran dari berbagai pihak. Oleh karena itu pada kesempatan ini penulis ingin mengucapkan terima kasih kepada pihak-pihak berikut:

1. Bapak Dr. Moch. BruriTriyono, selaku Dekan Fakultas Teknik Universitas Negeri Yogyakarta.
2. Bapak Muhammad Munir, M.Pd., selaku Ketua Jurusan Pendidikan Teknik Elektronika FT UNY.
3. Ibu Dr. Ratna Wardani, selaku Ketua Program Studi Pendidikan Teknik Informatika UNY.
4. Ibu Umi Rochayati, M.T., selaku dosen penasehat akademik.
5. Totok Sukardiyono, M.T., selaku dosen pembimbing skripsi.
6. Semua pihak yang telah membantu penyelesaian skripsi ini yang tidak dapat penulis sebutkan satu per satu.

Penulis menyadari masih banyak kekurangan dalam penyusunan skripsi ini, namun penulis tetap berharap skripsi ini bermanfaat bagi pengembangan ilmu pengetahuan terutama dalam kaitannya dengan pengembangan sistem pakar.

Yogyakarta, Oktober 2012

Penulis

Fajar Kurniawan

DAFTAR ISI

LEMBAR PERSETUJUAN.....	ii
LEMBAR PENGESAHAN	iii
HALAMAN PERNYATAAN	iv
MOTTO	v
PERSEMBAHAN	vi
ABSTRAK	vii
KATA PENGANTAR	viii
DAFTAR ISI.....	ix
DAFTAR LAMPIRAN	xii
DAFTAR TABEL.....	xiii
DAFTAR GAMBAR	xiv
BAB I PENDAHULUAN	1
A. Latar Belakang	1
B. Identifikasi Masalah	6
C. Batasan Masalah	6
D. Rumusan Masalah	7
E. Tujuan Penelitian	7
F. Manfaat Penelitian	8
BAB II KAJIAN PUSTAKA	9
A. Definisi Sistem	9
B. Definisi Software	10
1. Software Sistem Operasi	11
2. Software Aplikasi	12
3. Software Bahasa Pemrograman	12
4. Software Bantu	12
C. Definisi Web-based Software	13
D. PHP	14
E. MySQL	16
F. Software Quality	18

G. Rekayasa Perangkat Lunak	18
H. Analisis dan Perancangan Sistem	19
I. Model Pengembangan Air Terjun	20
J. Pemodelan Proses	21
1. Data Flow Diagram	21
K. Penyusunan Jadwal Pelajaran	23
1. Ant Colony Optimization	24
2. Algoritma Genetika	25
3. Blind Search	26
L. Pengujian Software	27
1. Pengujian Fungsional	28
2. Pengujian Usabilitas	29
3. Pengujian Antarmuka Pengguna	30
4. Pengujian Kompatibilitas	30
M. Penelitian yang Relevan	31
N. Kerangka Berfikir.....	32
O. Pertanyaan Peneliti.....	34
BAB III METODE PENELITIAN	35
A. Prosedur Pengembangan	35
1. Tahap Analisis.....	35
2. Tahap Perancangan	35
3. Tahap Pengembangan	36
4. Tahap Pengujian.....	36
B. Desain Penelitian	36
C. Definisi Operasional Variabel Penelitian	37
D. Instrumen dan Teknik Pengumpulan Data	38
1. Instrumen Penelitian	38
2. Uji Coba Instrumen	40
3. Teknik Pengumpulan Data	42
E. Teknik Analisis Data	42

BAB IV HASIL PENELITIAN DAN PEMBAHASAN	47
A. Pengembangan Perangkat Lunak	47
1. Analisis Kebutuhan	47
2. Perancangan Sistem	53
3. Pengembangan Sistem	57
4. Pengujian	60
a. Pengujian faktor <i>functionality</i>	60
b. Pengujian faktor <i>compatibility</i>	61
c. Pengujian faktor <i>user interface</i>	65
d. Pengujian faktor <i>usability</i>	66
B. Analisis Kualitas Perangkat Lunak	67
1. Analisis Faktor <i>user interface</i>	67
2. Analisis Faktor <i>functionality</i>	68
3. Analisis Faktor <i>compatibility</i>	70
4. Analisis Faktor <i>usability</i>	71
BAB V KESIMPULAN DAN SARAN	72
A. Kesimpulan	72
B. Keterbatasan Penelitian	72
C. Saran	73
D. Implikasi	73
DAFTAR PUSTAKA	74
LAMPIRAN	76

DAFTAR LAMPIRAN

Hasil uji validitas dan reliabilitas kuesioner	77
Hasil pengujian faktor functionality	79
Hasil pengujian faktor user interface	84
Hasil pengujian faktor compatibility web browser	89
Hasil pengujian faktor compatibility operating system	93
Hasil validasi kuesioner oleh dosen	94
Rekapitulasi data uji coba instrumen penelitian.....	97
Lembar kuesioner untuk ahli software	98
Hasil validasi perangkat lunak oleh dosen	103
Lembar kuesioner untuk operator	110
Rekapitulasi data hasil penelitian dan statistik deskriptif	114
Contoh hasil cetak jadwal pelajaran untuk mading.....	115
Contoh hasil cetak jadwal pelajaran untuk guru	116
Contoh hasil cetak jadwal pelajaran untuk kelas	117
Lembar Pengesahan Proposal Skripsi	118
Surat permohonan ijin penelitian	119
Surat keterangan telah melaksanakan penelitian	124

DAFTAR TABEL

Tabel 1 : Elemen-elemen dari DFD dan lambangnya	23
Tabel 2 : Format test case yang digunakan untuk pengujian	29
Tabel 3 : Kisi-kisi kuesioner kualitas perangkat lunak berbasis web	39
Tabel 4 : Standar pengembangan User Interface aplikasi berbasis web ...	43
Tabel 5 : Skor instrumen	45
Tabel 6 : Rincian skor maksimal minimal kuesioner	46
Tabel 7 : Pengkategorian skor instrumen	46
Tabel 8 : Pengkategorian skor total kuesioner	46
Tabel 9 : Pembagian hak akses perangkat lunak	54
Tabel 10 : Test case login-logout	60
Tabel 11 : Test case google chrome	62
Tabel 12 : Hasil pengujian operating system compatibility	63
Tabel 13 : Hasil pengujian printing compatibility	65
Tabel 14 : Test case screen font type	65
Tabel 15 : Rekapitulasi data kuesioner pengujian faktor usability	66
Tabel 16 : Rekapitulasi data kuesioner pengujian faktor user interface ...	67
Tabel 17 : Rekapitulasi data kuesioner pengujian faktor functionality.....	69
Tabel 18 : Rekapitulasi data kuesioner pengujian faktor compatibility....	70

DAFTAR GAMBAR

Gambar 1 : Bagan aplikasi berbasis web	13
Gambar 2 : Penyajian sederhana dari model pengembangan air terjun	20
Gambar 3 : Contoh data flow diagram pembayaran rekening listrik	21
Gambar 4 : Bagan alur penyelesaian algoritma ant colony optimation	25
Gambar 5 : Bagan alur penyelesaian algoritma genetika.....	26
Gambar 6 : Bagan alur penyelesaian algoritma blind search	27
Gambar 7 : Kerangka berfikir	33
Gambar 8 : Classic life cycle	35
Gambar 9 : Bagan alur penggunaan perangkat lunak	49
Gambar 10 : Struktur data jadwal pelajaran.....	52
Gambar 11 : Proses pencarian dalam penyelesaian jadwal pelajaran	53
Gambar 12 : DFD perangkat lunak	53
Gambar 13 : Tampilan tabel data guru.....	57
Gambar 14 : Tampilan form input data guru	58
Gambar 15 : Tampilan jadwal pelajaran utama	59
Gambar 16 : Tampilan pada google chrome	62
Gambar 17 : Tampilan pada windows 7	64
Gambar 18 : Perbandingan nilai kuesioner faktor kualitas user interface	68
Gambar 19 : Perbandingan nilai kuesioner faktor kualitas functionality..	69
Gambar 20 : Perbandingan nilai kuesioner faktor kualitas compatibility.	71
Gambar 21 : Perbandingan nilai kuesioner faktor kualitas usability	71

BAB I

PENDAHULUAN

A. Latar Belakang

Kemajuan yang sangat pesat di bidang teknologi, terutama teknologi informasi komputer, mendorong munculnya inovasi baru dalam penyajian dan pengolahan informasi untuk memenuhi kebutuhan informasi. Begitu juga dengan dunia pendidikan yang semakin melebarkan sayapnya dengan meningkatkan kapasitasnya, baik program studi, jumlah peserta didik, maupun jumlah guru pengajarnya. Sehingga secara langsung jumlah ruang kelas pun juga bertambah seiring dengan bertambahnya jurusan dan siswa. Di setiap sekolah, kegiatan yang rutin dilakukan setiap tahun pelajaran baru atau setiap semester adalah membuat/mengatur/menyusun jadwal pelajaran. Dalam pembuatannya memerlukan keseriusan kerja. Jika tidak teliti maka jadwal yang dibuat menjadi kurang sempurna yang berakibat pada guru yang akan mengajar maupun siswa yang belajar akan terjadi bentrok pertemuan kelas. Apalagi jumlah kelas dan guru yang banyak, sehingga tidak dipungkiri lagi diperlukan adanya perangkat lunak yang digunakan untuk membantu kegiatan ini.

Selama ini menyusun jadwal pelajaran dilakukan secara manual. Meskipun bisa saja terbantu dengan menggunakan program *spreadsheet* semacam MS Excel dengan menggunakan logika/*formula* yang disediakan program tersebut. Namun karena kurangnya pengalaman dengan *software MS Excel*, maka banyak sekali kesulitan yang akan dihadapi dalam penyusunan jadwal pelajaran tersebut. Di SMKN 1 Pacitan, berdasarkan hasil observasi penulis pada awal semester gasal

tahun 2011, pihak kurikulum melakukan penyusunan jadwal pelajaran ulang dan membutuhkan waktu sekurangnya 5 hari dengan bantuan *software spreadsheet MS Excel*. Kemudian karena masih ada beberapa pertemuan yang bentrok, maka dilakukan revisi jadwal pelajaran sekurangnya 2 kali revisi sehingga baru didapatkan jadwal pelajaran yang sempurna. Sehingga jumlah waktu yang dibutuhkan untuk menyusun jadwal pelajaran yang sempurna sekurangnya 7 hari. Kemudian pada awal semester genap tahun 2011 pihak kurikulum mulai menggunakan *software* yang lebih efektif dalam penyusunan jadwal pelajaran yaitu *ASC Timetables* dari website resmi <http://www.asctimetables.com>. Menurut pihak kurikulum SMKN 1 Pacitan yang menggunakannya, bapak Agus Prianggono, S.Si. mengatakan :

“Saya memakai program ini baru 2 semester dan saya merasa lebih mudah dalam menyusun jadwal pelajaran karena sudah ada fitur penjadwalan secara otomatis, namun tetap saja ada beberapa item yang harus saya atur lagi secara manual. Sayangnya di program ini tidak ada fitur pencarian pada saat entri data guru dan data mata pelajaran sehingga menyulitkan saya pada saat pencarian data.”

Pihak kurikulum juga menyampaikan adanya kesulitan untuk berbagi hak akses jika ada guru atau staff lain yang sekedar ingin melihat saja dari komputer mereka, sehingga harus mendistribusikan file secara manual melalui media *flashdisk*. Kemudian hal ini dapat diatasi dengan adanya sistem informasi sekolah yang ada, yaitu dengan menyediakan *link download* file dimana file yang disediakan merupakan hasil ekspor dari jadwal yang telah disusun pada *software*

ASC *timetables* dalam format HTML. Contoh dari hasil ekspor tersebut juga disediakan oleh website resmi di <http://www.asctt.com/ttexport/>.

Peneliti juga sempat melakukan observasi ke sekolah lain yaitu SMPN 4 Tulakan, Pacitan yang dalam menyusun jadwal pelajaran masih menggunakan aplikasi *spreadsheet MS Excel* hasil rekayasa salah satu guru yang sudah tidak ditugaskan di sekolah tersebut lagi. Dalam observasi tersebut peneliti menanyakan tentang kesulitan yang dihadapi dalam menyusun jadwal pelajaran dengan metode tersebut. Bapak Joko Priyanto, S.Pd selaku guru sekaligus staff kurikulum SMPN 4 Tulakan menyampaikan sebagai berikut :

“Sebenarnya sih sudah cukup mudah mas kalau membuat jadwal pelajaran dengan program ini karena format halaman cetak sudah tertata, tapi sayangnya karena di password maka saya tidak bisa menambahkan jumlah kelas lagi, padahal jumlah kelas di sekolah ini ada 7 setiap tingkatnya tapi yang disediakan program ini Cuma 5, trus hasil print-out nya juga tidak bisa berwarna, hanya ditulis kode guru saja, jadi kalau nyari jadwal harus nyari kode gurunya juga mas.”

Dari kutipan wawancara tersebut peneliti mendapat masukan bahwa pengguna ingin jumlah kelas yang dijadwalkan tidak hanya 5 untuk setiap tingkatnya, kemudian diharapkan hasil *print-out* nya juga bisa berwarna untuk memudahkan dalam membedakan satu guru dengan guru yang lainnya. Alasan lain peneliti memilih SMKN 1 Pacitan sebagai objek penelitian karena jumlah kelas dan gurunya lebih banyak dibandingkan sekolah lain, sehingga nantinya bisa diterapkan di sekolah lain.

Sistem penyusunan jadwal pelajaran ini dirancang untuk memenuhi kebutuhan pihak sekolah dalam menyusun jadwal pelajaran dengan berbagai fitur yang diminta oleh penggunanya yaitu fitur impor data, pencarian, ekspor data dan pembagian hak akses pengguna. Algoritma dan alur penggunaannya juga disesuaikan dengan kebutuhan pengguna. Selain itu sistem ini dirancang juga untuk mengembangkan sistem informasi yang ada mengingat di SMKN 1 Pacitan belum ada sistem informasi untuk jadwal pelajaran. Sehingga guru, staff dan siswa yang kompeten juga dapat ikut serta membangun sistem informasi tersebut agar lebih sempurna.

Di internet sudah tersedia begitu banyak pilihan *software* baik gratis maupun berbayar yang mendukung pembuatan jadwal pelajaran, namun pada implementasinya mengalami banyak kekurangan karena setiap sekolah memiliki strategi penjadwalan yang kompleks. Berdasarkan observasi penulis, di SMKN 1 Pacitan menggunakan strategi *rolling* ruang praktek, dan *rolling* waktu pagi/sore. Jumlah kelas ada 15 untuk tiap tingkat dengan total 45 kelas terdiri dari kelas 1, 2 dan 3. Maka dari itu sangat dibutuhkan sebuah perangkat lunak untuk menyusun jadwal pelajaran yang sesuai dengan kebutuhan sekolah tersebut. Perangkat lunak yang dirancang juga harus terdapat fitur yang dibutuhkan seperti impor data, pencarian, ekspor data dan pembagian hak akses pengguna sehingga akan lebih meningkatkan produktifitas sekolah.

Pada penelitian ini perangkat lunak yang dirancang terdapat perbedaan dengan perangkat lunak yang sudah ada, baik tampilan maupun strukturnya. Penulis menggunakan bahasa pemrograman PHP sebagai *interpreter*, MySQL

sebagai sistem penyimpanan data serta beberapa *plugin web* guna mendukung tampilan dan kemudahan penggunaannya seperti CSS, javascript, dan HTML. Sedangkan dasar fungsionalnya penulis melakukan observasi langsung dengan pihak terkait di SMKN 1 Pacitan. Karena fungsi utama dari sistem ini adalah menyusun jadwal pelajaran, maka peneliti menggunakan metode *tabu search* atau *blind search* yang diharapkan dapat membantu penggunaanya dalam menyusun jadwal pelajaran secara otomatis. Alasan peneliti menggunakan metode ini karena lebih mudah diimplementasikan dan terbukti tidak terjadi bentrok pertemuan untuk penjadwalan pelajaran.

Kemudian seperti halnya *software* lain, kualitas dari *software* ini juga harus diujikan dengan standar *SQA (Software Quality Assurance)* yaitu dengan mengukur kelayakan *software* dari berbagai faktor. Menurut salah satu situs yang menjelaskan tentang pengujian perangkat lunak berbasis web (<http://www.softwaretestinghelp.com/web-application-testing>) bahwa faktor pengujian dilakukan untuk perangkat lunak berbasis web diantaranya meliputi *Functionality, Usability, Interface, Compatibility, Performance, dan Security*. Sehingga perangkat lunak yang didapatkan dapat memberikan nilai lebih pada penggunaanya serta memiliki kesalahan maupun kekurangan yang minimal. Namun karena keterbatasan peneliti, maka untuk faktor *Security* dan *Performance* tidak dipaparkan pada penelitian ini.

B. Identifikasi Masalah

Berdasarkan latar belakang masalah yang telah dipaparkan, dapat diidentifikasi beberapa permasalahan antara lain :

1. Pihak kurikulum SMKN 1 Pacitan kesulitan dalam penyusunan jadwal pelajaran.
2. Penggunaan *software ASC Timetables* di SMKN 1 Pacitan saat ini masih terdapat beberapa kekurangan seperti pencarian data, impor data dan pembagian hak akses.
3. SMKN 1 Pacitan membutuhkan perangkat lunak penyusun jadwal pelajaran di sekolah untuk meningkatkan kinerja unit kurikulum.
4. Perlu adanya pengujian perangkat lunak dalam *Software* Sistem Penyusunan Jadwal Pelajaran Sekolah.
5. Perlu adanya analisis kualitas perangkat lunak dalam *Software* Sistem Penyusunan Jadwal Pelajaran Sekolah

C. Batasan Masalah

Dalam penyusunan tugas akhir ini diberikan batasan masalah agar dalam penjelasannya nanti akan lebih mudah, terarah dan sesuai dengan yang diharapkan. Batasan tersebut antara lain :

1. Sistem penjadwalan hanya berlaku untuk SMKN 1 Pacitan
2. Pengujian dan analisis kualitas hanya dilakukan pada 4 faktor yang diuji yaitu *functionality, compatibility, usability* dan *user interface*.

D. Rumusan Masalah

Berdasarkan masalah dari latar belakang yang ada maka rumusan masalah dari penelitian ini yaitu :

1. Bagaimana merancang software sistem penyusunan jadwal pelajaran sekolah berbasis web?
2. Seberapa baik kualitas *software* sistem penyusunan jadwal pelajaran sekolah berbasis web dari segi *Functionality*?
3. Seberapa baik kualitas *software* sistem penyusunan jadwal pelajaran sekolah berbasis web dari segi *User Interface*?
4. Seberapa baik kualitas *software* sistem penyusunan jadwal pelajaran sekolah berbasis web dari segi *Usability*?
5. Seberapa baik kualitas *software* sistem penyusunan jadwal pelajaran sekolah berbasis web dari segi *Compatibility*?

E. Tujuan Penelitian

Berdasarkan permasalahan yang diteliti, maka maksud dari penulisan tugas akhir ini adalah sebagai berikut :

1. Merancang *software* penyusun jadwal pelajaran sekolah berbasis web.
2. Memaparkan kualitas *software* sistem penyusunan jadwal pelajaran sekolah berbasis web dari faktor *Functionality*.
3. Memaparkan kualitas *software* sistem penyusunan jadwal pelajaran sekolah berbasis web dari faktor *User Interface*.
4. Memaparkan kualitas *software* sistem penyusunan jadwal pelajaran sekolah berbasis web dari faktor *Usability*

5. Memaparkan kualitas *software* sistem penyusunan jadwal pelajaran sekolah berbasis web dari faktor *Compatibility*.

F. Manfaat Penelitian

Adapun manfaat yang diinginkan dari penelitian adalah :

1. Membantu unit kurikulum sekolah dalam menyusun jadwal pelajaran sekolah.
2. Mendapatkan hasil analisis kualitas perangkat lunak sistem penyusunan jadwal pelajaran sekolah berbasis web
3. Perangkat lunak yang dibuat dapat dijadikan referensi dalam mengembangkan perangkat lunak sejenis berikutnya.

BAB II

KAJIAN PUSTAKA

A. Definisi Sistem

Sistem merupakan suatu susunan yang teratur dari kegiatan-kegiatan yang saling bergantung dan prosedur-prosedur yang berhubungan yang melaksanakan dan memudahkan pelaksanaan kegiatan utama dari suatu organisasi. Sejak orang mengenal tentang sistem, maka banyak definisi yang telah dikemukakan. Beberapa ahli mendefinisikan mengenai sistem sebagai berikut:

Hanif Al Fatta (2007:2) mendefinisikan “secara sederhana sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsure atau variabel-variabel yang saling terorganisasi, saling berinteraksi, dan saling bergantung sama lain.”

James A. O'Brien, George M. Marakas (2010:26) mendefinisikan sistem sebagai seperangkat komponen yang saling terkait, dengan batas yang jelas, bekerja sama untuk mencapai seperangkat tujuan dengan menerima input dan menghasilkan output produksi dalam suatu proses transformasi yang terorganisir.

Jack J. Champlain (2003:15) juga mendefinisikan bahwa sistem adalah aplikasi perangkat lunak komputer yang melakukan fungsi bisnis, mendukung sistem manajemen basis data, menyediakan akses perangkat keras dimana ia berada, dan sistem operasi yang mengatur perangkat keras. Phillip A. Laplante (2004:2) mengartikan sebuah sistem adalah pemetaan satu set input menjadi satu set output.

Berdasarkan pendapat para ahli di atas, dapat disimpulkan sistem adalah serangkaian komponen, prosedur yang saling bekerjasama, terkait, terpadu, dan

saling berhubungan satu sama lain secara harmonis membentuk satu kesatuan, untuk mencapai tujuan atau sasaran yang spesifik.

B. Definisi Software

Software secara umum dapat diartikan sebagai sekumpulan data-data elektronik yang tersimpan dan diatur oleh komputer yang berupa program atau instruksi untuk menjalankan dan mengeksekusi suatu perintah. *Software* merupakan perangkat yang ada di dalam komputer yang tidak dapat kita sentuh dan tidak dapat kita lihat bentuk fisiknya, namun dapat kita operasikan saat kita menggunakan komputer. Sedangkan secara khusus *software* diartikan sebagai “sekumpulan perintah/instruksi yang jika dijalankan akan menghasilkan fungsi dan hasil yang diinginkan” (RogerPressman,2001:6).

B.B. Agarwal, S.P. Tayal dan M. Gupta (2010:4) dalam bukunya yang berjudul *Software Engineering and Testing* menyebutkan “Software is a set of instructions used to acquire inputs and to manipulate them to produce the desired output in terms of functions and performance as determined by the user of the software”. Kutipan tersebut menyebutkan bahwa *software* adalah sekumpulan instruksi yang digunakan untuk memperoleh masukan (input) dan memanipulasinya sehingga menghasilkan keluaran (output) yang diinginkan dalam hal fungsi dan kinerja yang ditentukan oleh pengguna *software*. Beberapa contoh dari *software* adalah Adobe Photoshop, menyediakan fungsi untuk mengolah gambar, Microsoft Word, menyediakan fungsi untuk mengolah dokumen.

Berdasarkan fungsinya, *software* dibedakan menjadi : *Software* Sistem Operasi (Operating System), *Software* Aplikasi (Application), *Software* Bahasa Pemrograman (Programming Language), dan *Software* Program Bantu (Utility). Berikut penjelasan dari masing-masing jenis *software*.

1. *Software* Sistem Operasi (*Operating System*)

Software Sistem Operasi yaitu program yang berfungsi untuk mengendalikan sistem kerja yang mendasar sehingga mengatur kerja media input, output, tabel pengkodean, memori, penjadwalan prosesor, dan lain-lain. Sistem operasi berfungsi sebagai penghubung antara manusia dengan perangkat keras dan perangkat lunak yang akan digunakan. Adapun fungsi utama sistem operasi adalah

- a. Menyimpan program dan aksesnya
- b. Membagi tugas di dalam CPU
- c. Mengalokasikan tugas-tugas penting
- d. Merekam sumber-sumber data
- e. Mengatur memori sistem termasuk penyimpanan, menghapus dan mendapatkan data
- f. Memeriksa kesalahan sistem
- g. Multitugas pada OS/2", Windows '95", Windows '98", Windows NT", /2000/XP
- h. Memelihara keamanan sistem, khusus pada jaringan yang membutuhkan kata sandi (password) dan penggunaan ID

Contoh *Software* Sistem Operasi, misalnya : Disk Operating System (DOS), Microsoft Windows, Linux, dan Unix.

2. *Software Aplikasi (Application Software)*

Software Aplikasi adalah perangkat lunak yang dirancang khusus untuk kebutuhan tertentu, misalnya program pengolah kata, mengelola lembar kerja, program presentasi, design grafis, dan lain-lain.

3. *Software Bahasa Pemrograman (Programming Language Software)*

Software bahasa pemrograman yaitu program yang digunakan untuk menerjemahkan instruksi-instruksi yang ditulis dalam bahasa pemrograman ke bahasa mesin dengan aturan atau prosedur tertentu, agar diterima oleh komputer.

Ada 3 level bahasa pemrograman, yaitu :

a. Bahasa tingkat rendah (low level language)

Bahasa ini disebut juga bahasa mesin (assembler), dimana pengkodean bahasanya menggunakan kode angka 0 dan 1.

b. Bahasa tingkat tinggi (high level language)

Bahasa ini termasuk dalam bahasa pemrograman yang mudah dipelajari oleh pengguna komputer karena menggunakan bahasa Inggris. Contohnya : BASIC, COBOL, PASCAL, FORTRAN.

c. Bahasa generasi keempat (4 GL)

Bahasa pemrograman 4 GL (Fourth Generation Language) merupakan bahasa yang berorientasi pada objek yang disebut Object Oriented Programming (OOP). Contoh software ini adalah : PHP, Java, Visual NET

4. *Software Bantu (Utility)*

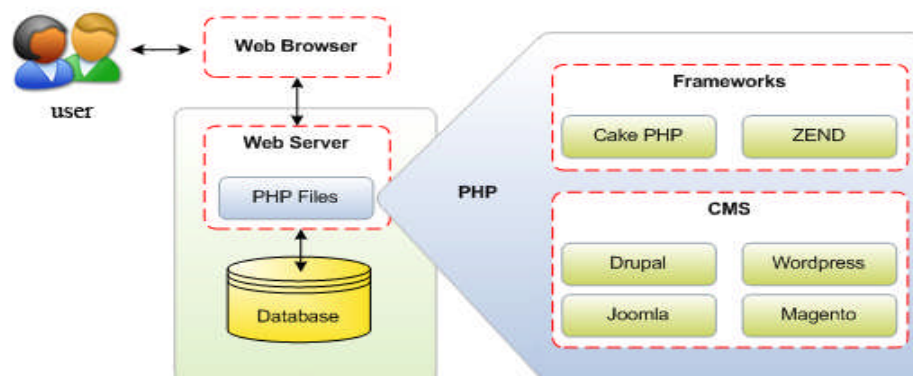
Software bantu merupakan perangkat lunak yang berfungsi sebagai aplikasi pembantu dalam kegiatan yang ada hubungannya dengan komputer, misalnya

memformat disket, mengopi data, mengkompres file, dan lain-lain. Contoh software ini diantaranya : Norton Utility, Winzip, Norton Ghost, Antivirus. Pada penelitian ini, *software* yang dibuat merupakan *software* aplikasi karena tujuannya adalah untuk membantu penyusunan jadwal pelajaran sekolah.

C. Definisi Web-based software

Salah satu jenis dari software adalah *web-based software*. Dikutip dari buku Software Engineering by Roger Pressman 2001 halaman 10 “*The Web pages retrieved by a browser are software that incorporates executable instructions (e.g., CGI, HTML, Perl, or Java), and data (e.g., hypertext and a variety of visual and audio formats)*”. Dari kutipan tersebut dijelaskan bahwa software berbasis web merupakan penggabungan dari instruksi *interpreter* seperti CGI, Perl, atau Java dengan data, baik *hypertext* atau data lainnya.

Dalam penelitian ini, software yang dibuat adalah kombinasi dari PHP sebagai *interpreter* dengan MySQL sebagai *database* yang kemudian ditampilkan melalui *browser* dengan format HTML. Secara umum arsitektur aplikasi berbasis web digambarkan sebagai berikut.



Gambar 1 : Bagan aplikasi berbasis web

Pada gambar diatas dijelaskan bahwa berkas PHP disimpan pada sebuah *web server* yang secara langsung melayani permintaan (*request*) dari *web browser* oleh pengguna (*user*). *Web server* mengolah *script* PHP tersebut sekaligus melakukan transaksi data dengan *database* dan kembali menampilkan hasilnya pada *web browser* pengguna. Berkas PHP dapat berupa *frameworks* maupun CMS (*Content Managemen System*). *Web server* dapat menggunakan apache atau IIS (*Internet Information Service*) yang dikembangkan oleh Microsoft.

Sedangkan *database server* dapat menggunakan MySQL atau MsSQL. Untuk dapat menggunakan aplikasi pengguna dapat mengakses melalui *web browser* misalnya Mozilla Firefox dengan memanggil url dari *web server*. Arsitektur software berbasis web tersebut dapat dijalankan pada berbagai sistem operasi, baik *windows* atau *linux* maupun *OS X*. Pada penelitian ini penulis menggunakan sistem operasi *Microsoft Windows XP* karena disesuaikan dengan kondisi di sekolah yang diobservasi.

D. PHP

“PHP is a widely-used general-purpose scripting language that is especially suited for Web development and can be embedded into HTML.” Salah satu kutipan dari situs resmi PHP (<http://www.php.net>) yang menjelaskan bahwa PHP merupakan salah satu bahasa pemrograman yang dapat digunakan secara luas dalam membangun *software* berbasis web yang sekaligus dapat diintegrasikan dengan HTML. Selain itu lebih dari 20 juta website di dunia menggunakan PHP sebagai interpretnya pada akhir 2006 dan terus bertambah hingga saat ini (Vikram Vaswani,2007). Kedua alasan tersebut yang menjadikan

penulis memilih PHP sebagai interpreter dalam penelitian ini. Dikutip dari situs resmi PHP (<http://www.php.net/manual/en/intro-whatcando.php>) beberapa kelebihan yang terdapat dalam PHP antara lain :

1. PHP dapat digunakan pada pemrograman sisi server (server side scripting).
2. Pemrograman PHP dapat dilakukan secara *command line* tanpa adanya server dan web browser.
3. Pemrograman PHP dapat dilakukan secara prosedural, object oriented, atau kombinasi dari keduanya.
4. Web server yang mendukung PHP dapat ditemukan dimana-mana dari mulai ISS maupun apache dengan konfigurasi yang relative mudah.
5. PHP adalah bahasa pemrograman *open source* yang dapat digunakan di berbagai mesin (linux, unix, apple, windows) dan dapat dijalankan secara runtime melalui *console* serta melalui *web browser*.
6. Hasil keluaran PHP tidak hanya HTML tapi dapat menghasilkan file PDF, file gambar, XML dan format file lainnya yang dibuat secara otomatis melalui PHP.
7. PHP mendukung berbagai jenis database seperti MySQL, MsSQL, maupun ODBC.
8. PHP juga memiliki dukungan untuk berkomunikasi dengan layanan lain menggunakan protokol seperti LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (pada Windows) dan lain-lain.

Serta masih banyak lagi kelebihan yang terdapat dalam PHP yang tidak bisa penulis paparkan semuanya di penelitian ini.

E. MySQL

Salah satu kebutuhan dari software yang akan dibuat adalah menyimpan data, dimana semua data dari jadwal yang disusun akan disimpan pada database, baik sebelum jadi maupun setelah jadi. Database merupakan tempat penyimpanan data (angka dan huruf) dari sebuah sistem. Sedangkan penulis menggunakan salah satu system database yang ada yaitu MySQL. Seperti yang dikutip dari buku *Beginning MySQL*, “*MySQL is a relational database management system (RDBMS), allows you to manage data and support applications that rely on access to MySQL data*” (Rober Sheldon and Geoff Moes,2005). Dijelaskan bahwa MySQL merupakan sebuah sistem manajemen database yang ter-relasi yang dapat digunakan untuk mendukung aplikasi yang membutuhkan *database*.

Penulis memilih MySQL karena pada software yang akan dibuat melibatkan struktur hirarki antara satu variabel dengan variabel lainnya. Seperti hubungan antara guru dengan pelajaran, hari dengan jam pelajaran, serta variabel lain yang harus dipenuhi guna mendukung pembuatan software ini. Berikut ini adalah kelebihan dari MySQL menurut Rober Sheldon and Geoff Moes,(2005:8) dalam bukunya yang berjudul *Beginning MySQL*

1. MySQL adalah *database* yang memiliki kecepatan yang tinggi dalam melakukan pemrosesan data, dapat diandalkan, dan mudah digunakan serta mudah dipelajari.
2. *Fully Multi Threaded* dengan *kernel thread* artinya adalah bisa dengan mudah mempergunakan *multiple CPU* bila ada.

3. API (*Application Programming Interface*) dengan C, C++ , Eiffel, Java, Perl, PHP, Python dan Tel
4. MySQL mendukung banyak bahasa pemrograman seperti C, C++, Perl, Python, Java, dan PHP. Selain itu, dengan bantuan ODBC, MySQL juga mampu berinteraksi dengan berbagai pemrograman visual seperti Delphi, Visual Basic, Java, dan sebagainya.
5. MySQL dapat melakukan koneksi dengan client menggunakan *protocol* TCP/IP, Unix socket (Unix), atau Named Pipes (NT).
6. MySQL dapat menangani data dengan skala yang sangat besar dengan jumlah *record* mencapai lebih dari 50 juta, menampung 60 ribu tabel, dan juga bisa menampung 5 milyar baris data.
7. Dalam hal relasi antartabel MySQL menerapkan metode *onesweep multijoin*, sehingga sangat efisien dalam mengelola informasi yang kita minta dari beberapa tabel sekaligus
8. *Multiuser*, yaitu dalam satu database server pada MySQL dapat diakses oleh beberapa user dalam waktu yang sama tanpa mengalami konflik atau kemacetan sistem.
9. *Security* yang dimiliki *database* MySQL dikenal baik, karena memiliki lapisan keamanan seperti level *subnetmask*, nama *host* dan izin akses *user* dengan sistem perizinan yang khusus serta *password* yang dimiliki setiap *user* dalam bentuk data terenkripsi

F. SQ (Software Quality)

Software Quality didefinisikan sebagai kesesuaian yang diharapkan pada semua perangkat lunak yang dibangun berkaitan dengan fungsi perangkat lunak, standar pembangunan perangkat yang terdokumentasi, dan karakteristik yang ditujukan oleh perangkat lunak (B.B. Agarwal, S.P. Tayal, M. Gupta, 2010:89). Definisi ini menekankan pada 3 hal, yaitu:

1. Kebutuhan perangkat lunak adalah dasar ukuran kualitas perangkat lunak, jika perangkat lunak tidak sesuai dengan kebutuhan yang ditentukan, maka kualitasnya pun berkurang.
2. Jika menggunakan sebuah standar untuk pembangunan perangkat lunak, maka perangkat lunak dianggap kurang berkualitas jika tidak memenuhi standar tersebut.
3. Sering kali ada kualitas yang secara langsung tertulis, seperti kemudahan penggunaan dan pemeliharaan yang baik. Sehingga kualitas perangkat lunak dipertanyakan jika tidak memenuhi kebutuhan ini.

G. Rekayasa Perangkat Lunak

Rekayasa atau teknik adalah penerapan ilmu dan teknologi untuk menyelesaikan permasalahan manusia. Hal ini diselesaikan lewat pengetahuan, matematika, dan pengalaman praktis yang diterapkan untuk mendesain objek atau proses yang berguna (Janner Simarmata, 2010:10). Rekayasa perangkat lunak (RPL atau [*Software Engineering*]) adalah satu bidang profesi yang mendalami cara-cara pengembangan perangkat lunak termasuk pembuatan, pemeliharaan, manajemen organisasi pengembangan perangkat lunak dan sebagainya.

The software engineering body of knowledge (2004:2) membagi rekayasa perangkat lunak ke dalam 10 area pengetahuan, yaitu :

1. Kebutuhan perangkat lunak
2. Perancangan perangkat lunak
3. Konstruksi perangkat lunak
4. Pengujian perangkat lunak
5. Pemeliharaan perangkat lunak
6. Manajemen konfigurasi perangkat lunak
7. Manajemen perangkat lunak
8. Proses perangkat lunak
9. Metode dan tool perangkat lunak, dan
10. Kualitas perangkat lunak.

H. Analisis dan Perancangan Sistem.

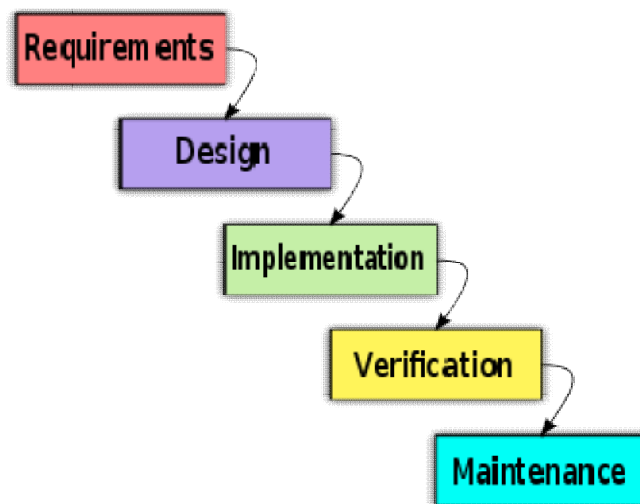
Analisis sistem didefinisikan sebagai bagaimana memahami dan menspesifikasi dengan detail apa yang harus dilakukan oleh sistem. Sementara sistem desain diartikan sebagai menjelaskan dengan detail bagaimana bagian-bagian dari sistem diimplementasikan. Sehingga analisis dan perancangan sistem (ANSI) bisa didefinisikan sebagai : Proses organisasional kompleks dimana sistem informasi berbasis komputer diimplementasikan (Hanif Al Fatta, 2007:24).

Tahapan-tahapan dalam ANSI menjadi sangat penting karena kesuksesan suatu sistem informasi tergantung pada analisis dan perancangan yang baik. Tahapan analisis akan menentukan masalah apa yang harus diselesaikan pada organisasi atau perusahaan. Kesalahan pada tahap ini akan mengakibatkan

masalah tetap ada walaupun sistem informasi telah diimplementasikan. Sementara tahapan desain akan sangat menentukan seperti apa sistem akan berfungsi. walaupun pada tahapan analisis masalah utama sudah terpetakan dengan benar, kesalahan desain akan mengakibatkan kegagalan penyelesaian masalah oleh sistem komputer.

I. Model Pengembangan Air Terjun

Salah satu model pengembangan yang cukup populer adalah model air terjun (*waterfall model*). Model pengembangan ini memacu pengembang untuk merinci apa yang seharusnya perangkat lunak lakukan (mengumpulkan dan menentukan kebutuhan sistem) sebelum sistem tersebut dikembangkan.



Gambar 2 : Penyajian sederhana dari model pengembangan air terjun

Model ini memungkinkan memecah misi pengembangan yang rumit menjadi beberapa langkah logis (desain, kode, pengujian, dan seterusnya) dengan beberapa langkah yang pada akhirnya akan menjadi produk akhir yang siap pakai. Untuk memastikan bahwa sistem bisa dijalankan, setiap langkah akan membutuhkan validasi, masukan, dan kriteria yang ada (Janner Simarmata, 2010:54).

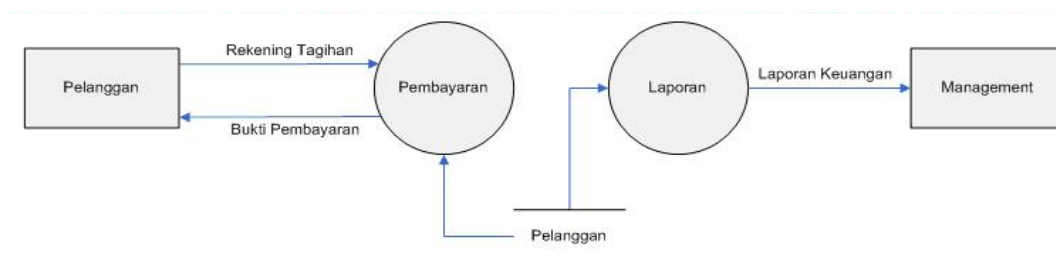
J. Pemodelan Proses

Pemodelan proses adalah cara formal untuk menggambarkan bagaimana bisnis beroperasi. Mengilustrasikan aktivitas-aktivitas yang dilakukan dan bagaimana data berpindah di antara aktivitas-aktivitas itu. Ada banyak cara untuk merepresentasikan proses model. Salah satu cara yang populer adalah dengan menggunakan data flow diagram (DFD). Ada dua jenis DFD, yaitu DFD logis dan DFD fisik. DFD logis menggambarkan proses tanpa menyarankan bagaimana mereka akan dilakukan, sedangkan DFD fisik menggambarkan proses model berikut implementasi pemrosesan informasinya.

1. Data Flow Diagram

Data Flow Diagram (DFD) adalah suatu teknik grafik yang digunakan untuk menggambarkan aliran informasi dan perpindahan data dari input ke output-nya (Pressman, 2001:311). DFD biasa dikenal dengan data flow graph atau bubble chart.

Data Flow Diagram menunjukkan aliran data antar komponen. Komponen dapat berupa tugas, komponen perangkat lunak, atau bahkan gambaran dari sebuah fungsi yang dimasukkan dalam perangkat lunak (David A. Gustafson, 2002:9). Berikut adalah contoh DFD Pembayaran rekening listrik.



Gambar 3 : Contoh Data Flow Diagram Pembayaran rekening listrik

Untuk dapat membaca suatu DFD kita harus memahami dulu, elemen-elemen yang menyusun suatu DFD. Menurut Hanif Al Fatta (2007:106) ada empat elemen yang menyusun suatu DFD, yaitu :

a. Proses.

Aktivitas atau fungsi yang dilakukan untuk alasan bisnis yang spesifik, biasa berupa manual maupun terkomputasi.

b. Data flow

Satu data tunggal atau kumpulan logis suatu data, selalu diawali atau berakhir pada suatu proses.

c. Data store

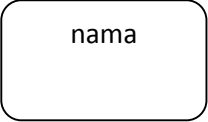
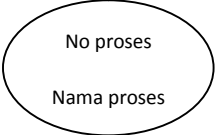
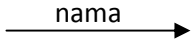
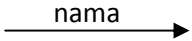


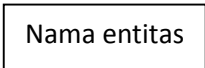
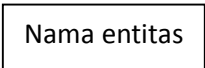
Kumpulan data yang disimpan dengan cara tertentu. Data yang mengalir disimpan dan data store. Aliran data di-update atau ditambahkan ke data store

d. External entity

Orang, organisasi, atau sistem yang berada di luar sistem tetapi berinteraksi dengan sistem.

Masing-masing elemen akan diberi lambang tertentu untuk membedakan satu dengan yang lain. Ada beberapa metode untuk menggambarkan elemen-elemen tersebut. Untuk lebih jelasnya dapat dilihat pada tabel gambar berikut:

Tabel 1 : Elemen – elemen dari DFD dan lambangnya

Elemen data flow diagram	Field tipikal yang biasa digunakan	Simbol Gene And Sarson	Simbol De Marco And Jourdan
Setiap proses memiliki nomor, nama, deskripsi proses, satu/lebih output data flow, satu/lebih input flow	Label (nama) Type (proses) Deskripsi Nomor proses		
Setiap data flow memiliki nama, deskripsi, satu/lebih koneksi ke suatu proses	Label Type Deskripsi Alias Komposisi (deskripsi dari elemen-elemen data)		
Setiap data store memiliki nomor, nama, deskripsi, satu/lebih input data flow, satu/lebih output data flow.	Label (nama) Type Deskripsi Alias Komposisi Catatan		
Setiap entitas memiliki nama, deskripsi	Label Type Deskripsi Alias Deskripsi entitas		

K. Penyusunan Jadwal Pelajaran

Pengertian jadwal menurut kamus besar bahasa Indonesia adalah pembagian waktu berdasarkan rencana pengaturan urutan kerja; daftar atau tabel kegiatan atau rencana kegiatan dengan pembagian waktu pelaksanaan yang terperinci. Sedangkan pengertian penyusunan jadwal atau penjadwalan adalah proses, cara, perbuatan menjadwalkan atau memasukkan ke dalam jadwal

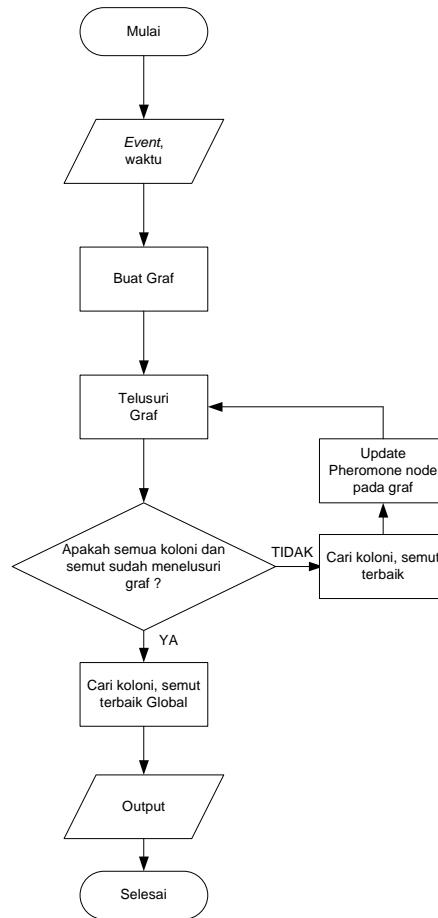
(<http://bahasa.kemdiknas.go.id>). Tujuan utama dari sebuah sekolah adalah terselenggaranya kegiatan belajar mengajar (KBM). Adapun beberapa komponen utama dari KBM adalah tersedianya guru pengajar, ruang kelas, siswa, materi pembelajaran, dan waktu KBM. Maka dari itu dalam menyelenggarakan KBM membutuhkan pengaturan komponen-komponen tersebut secara rinci yang disebut jadwal pelajaran.

Secara umum definisi penyusunan jadwal pelajaran sekolah adalah mengatur atau membagi komponen KBM sehingga dapat terlaksana proses KBM dengan baik. Penyusunan jadwal pelajaran sekolah dapat juga diartikan sebagai kegiatan mengalokasikan sejumlah mata pelajaran beserta guru pengajar ke sejumlah kelas yang tersedia dan sejumlah *slot* waktu disertai dengan *constraint*. *Constraints* terbagi atas dua jenis, yaitu *hard constraints* dan *soft constraints* (Petrovic dan Burke, 2004:2). *Hard constraints* merupakan batas-batas yang harus diterapkan pada penjadwalan dan harus dipenuhi. *Soft constraints* didefinisikan sebagai batas-batas mengenai alokasi sumber daya yang jika dilanggar masih dapat menghasilkan solusi yang layak tetapi sedapat mungkin untuk dipenuhi. Oleh sebab itu terdapat beberapa cara untuk memenuhi *constraint* tersebut.

1. Ant Colony Optimization.

Pernah diteliti oleh Yana Maulana (Maulana, 2005) dalam penelitiannya yang berjudul “penjadwalan perkuliahan dengan menggunakan algoritma *max-min ant system*”. Ant Colony Optimization (ACO) pertama kali diperkenalkan oleh Marco Dorigo, ACO itu sendiri terinspirasi oleh koloni-koloni semut dalam mencari makan. Semut-semut tersebut meninggalkan zat (*pheromone*) di jalan

yang mereka lalui. Algoritma ACO ini merupakan algoritma pencarian berdasarkan probabilistik, di mana probabilistik yang digunakan merupakan probabilistik dengan bobot sehingga butir pencarian dengan bobot yang lebih besar akan berakibat memiliki kemungkinan terpilih yang lebih besar pula. Secara umum alur penyelesaian dengan ACO digambarkan sebagai berikut :

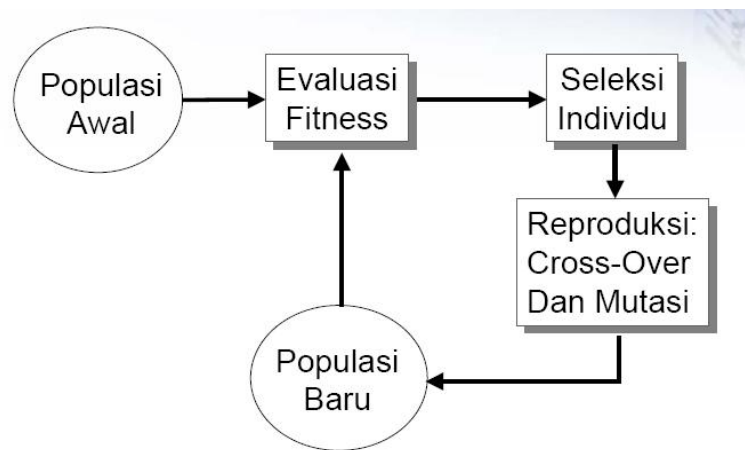


Gambar 4 : Bagan alur penyelesaian algoritma ant colony optimization

2. Algoritma Genetika

Algoritma Genetika adalah algoritma yang memanfaatkan proses seleksi alamiah yang dikenal dengan proses evolusi. Dalam proses evolusi, individu secara terus-menerus mengalami perubahan gen untuk menyesuaikan dengan

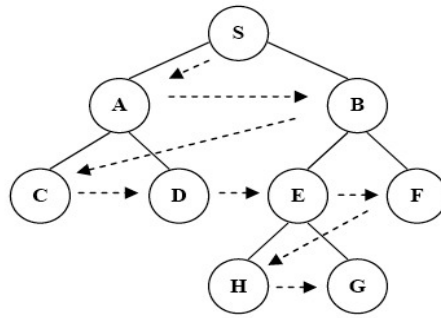
lingkungan hidupnya. “Hanya individu-individu yang kuat yang mampu bertahan”. Proses seleksi alamiah ini melibatkan perubahan gen yang terjadi pada individu melalui proses perkembangbiakan. Dalam algoritma genetika ini, proses perkembang-biakan ini menjadi proses dasar yang menjadi perhatian utama, dengan dasar berpikir “Bagaimana mendapatkan keturunan yang lebih baik”. Algoritma genetika ini ditemukan oleh John Holland dan dikembangkan oleh muridnya David Goldberg. Secara umum algoritma genetika menyelesaikan permasalahan dengan alur sebagai berikut:



Gambar 5 : Bagan alur penyelesaian algoritma genetika

3. Pencarian Buta (*Blind Search*)

Pencarian buta atau lebih dikenal dengan *Blind Search* merupakan metode pencarian konvensional yang mudah digunakan. Metode pencarian ini akan melewati setiap titik yang ada sampai menemukan solusi yang ditentukan. Proses pencarian buta dapat digambarkan seperti berikut :



Gambar 6 : Bagan alur penyelesaian algoritma Blind Search

Pada gambar tersebut dijelaskan bahwa semua node pada level n akan dikunjungi terlebih dahulu sebelum level $n+1$, mulai dari akar terus ke level 1 dari kiri ke kanan, kemudian ke level selanjutnya hingga solusi ditemukan.

Dalam penyusunan jadwal pelajaran metode pencarian digunakan untuk menemukan *slot* waktu yang tidak melanggar *constraint* yang ditentukan sehingga setiap guru dan kelas dapat melaksanakan KBM tanpa bentrok dengan guru atau kelas lain pada waktu bersamaan.

L. Pengujian Software

Pengujian perangkat lunak adalah proses untuk menjalankan sebuah program atau sistem untuk mencari kesalahan. Pentingnya pengujian perangkat lunak dan implikasinya mengacu pada kualitas perangkat lunak yang tidak dapat terlalu ditekan karena melibatkan sederetan aktivitas produksi dengan peluang terjadi kesalahan manusia yang sangat besar dan ketidakmampuan manusia untuk melakukan dan berkomunikasi dengan sempurna. Oleh karena itu, pengembangan perangkat lunak diiringi dengan aktivitas jaminan kualitas. Proses pengujian memiliki kecenderungan untuk memandang perangkat lunak dari sisi kebutuhan pengguna sehingga takaran yang diukur merupakan kesesuaian dan kemampuan

perangkat lunak untuk memenuhi kebutuhan pengguna akhir. Mengacu pada standar pengujian perangkat lunak berbasis web yang terdapat pada website (<http://www.softwaretestinghelp.com/web-application-testing>), terdapat 6 macam pengujian yaitu :

1. Functionality testing
2. Usability testing
3. Interface testing
4. Compatibility testing
5. Performance testing
6. Security testing

Ke-6 pengujian tersebut dilakukan untuk mengukur seberapa baik kualitas sebuah perangkat lunak berbasis web. Pada penelitian ini pengujian difokuskan pada 4 hal yang mendukung kualitas dari perangkat lunak yang dikembangkan. 4 hal tersebut antara lain :

1. Pengujian fungsional (*functionality testing*)

Pada jenis pengujian ini, perangkat lunak diuji untuk persyaratan fungsional. Pengujian dilakukan dalam bentuk tertulis untuk memeriksa apakah aplikasi berjalan sesuai yang diharapkan. Walaupun pengujian fungsional sudah sering dilakukan di bagian akhir dari siklus pengembangan, masing-masing komponen dan proses dapat diuji pada awal pengembangan, bahkan sebelum sistem berfungsi, pengujian ini sudah dapat dilakukan pada seluruh sistem. Pengujian fungsional meliputi seberapa baik sistem melaksanakan fungsinya, termasuk perintah-perintah pengguna, manipulasi data, pencarian dan proses bisnis,

pengguna layar, dan integrasi. Pengujian fungsional juga meliputi permukaan yang jelas dari jenis fungsi-fungsi, serta operasi *back-end* (seperti, keamanan dan bagaimana meningkatkan system). Proses pengujian fungsional dilakukan dengan metode *black box* menggunakan *test case* untuk memastikan bahwa setiap fungsi utama dari sistem berjalan sesuai tugasnya, perangkat lunak dianggap gagal jika ada minimal satu fungsi utama yang tidak berjalan. Setiap fungsi yang diujikan ditulis pada format *test case* seperti contoh berikut ini.

Tabel 2 : Format *test case* yang digunakan untuk pengujian

<i>Test case id</i>	Nama yang unik untuk idektifikasi test case
<i>Purpose</i>	Tujuan dari test case
<i>Assumptions</i>	Syarat kondisi awal yang harus terpenuhi sebelum test dapat dijalankan.
<i>Test data</i>	Variabel atau kondisi yang akan di test.
<i>Steps</i>	Langkah – langkah yang dijalankan.
<i>Expected result:</i>	Hasil yang seharusnya didapatkan (yang menunjukkan bahwa tidak ada kesalahan dalam perangkat lunak)
<i>Actual result:</i>	Hasil yang didapat dalam pengujian.
<i>Pass/Fail:</i>	Keterangan : Lolos atau Gagal.

2. Pengujian usabilitas (*usability testing*).

Pengujian ini disebut juga sebagai pengujian untuk keakraban pengguna (*testing for user-friendliness*). Pengujian ini dilakukan jika antarmuka pengguna dari aplikasinya penting dan harus spesifik untuk jenis pengguna tertentu. Pengujian usabilitas adalah proses yang bekerja dengan pengguna akhir secara langsung maupun tidak langsung untuk menilai bagaimana pengguna merasakan paket perangkat lunak dan bagaimana mereka berinteraksi dengannya. Dalam penelitian ini pengujian usabilitas dilakukan menggunakan kuesioner.

3. Pengujian Antarmuka Pengguna (*user interface testing*)

Pengujian ini dilakukan untuk menilai apakah perangkat lunak yang dibuat sudah memenuhi beberapa prinsip antarmuka pengguna. Hanif Al Fatta (2007:153) dalam bukunya yang berjudul Analisis & Perancangan Sistem Informasi menyebutkan beberapa prinsip antarmuka pengguna yaitu:

- a. Antarmuka yang baik tidak mengharuskan pengguna untuk mengingat tampilan antarmuka pengguna.
- b. Antarmuka pengguna menampilkan apa yang dimengerti oleh pengguna atau visualisasi keadaan dari system sekarang.

Selain itu ada beberapa hal yang harus dihindari dalam desain antarmuka, yaitu :

- a. Menampilkan terlalu banyak informasi dan terlalu banyak pilihan.
- b. Menampilkan terlalu sedikit informasi, terlalu sedikit pilihan, dan tanpa konteks.
- c. Eksploitasi struktur menu standar yang sudah familiar dengan perangkat lunak yang sering digunakan pengguna.

4. Pengujian kompatibilitas (*compatibility testing*)

Pengujian ini menjadi sangat penting dalam konteks perangkat lunak berbasis web, mengingat penggunaannya akan melalui *browser* yang jenis ragamnya begitu banyak. Beberapa perangkat lunak berbasis web memiliki ketergantungan pada browser tertentu, sehingga dibutuhkan pengujian untuk mengetahui seberapa cocok perangkat lunak dapat dijalankan pada berbagai browser yang mungkin dipakai oleh pengguna. Beberapa contoh browser yang

sering digunakan seperti Internet Explorer, Mozilla firefox, google chrome dan opera.

Sebelum dilakukan pengujian pada pengguna akhir, penulis melakukan validasi perangkat lunak terlebih dahulu kepada 3 dosen yang berkompeten dalam rekayasa perangkat lunak. Ketiga dosen tersebut adalah Yuniar Indriharsari, S.T., M.Eng, Dr. Drs. Eko Marpanaji, M.T, dan Drs. Herman Dwi Surjono, M.Sc, M.T, Ph.D yang ketiganya adalah dosen Teknik Informatika Fakultas Teknik UNY. Validasi meliputi faktor fungsionalitas, antarmuka pengguna dan kompatibilitas dengan kuesioner yang telah disiapkan.

M. Penelitian yang Relevan

1. Skripsi, Cecep Susan Jumena, 2006, Penyelesaian Masalah Penjadwalan Kuliah Menggunakan Algoritma Tabusearch

(Jumena, 2006) Dalam penelitiannya menyebutkan bahwa hasil penjadwalan dengan algoritma *tabu search* tidak terjadi bentrok antar komponen yang dijadwalkan yaitu kelas, matakuliah, dosen, waktu, dan ruang. Hal ini karena pertimbangan dari beberapa *constraint* yang dihitung secara matematis dari algoritma tersebut. Sehingga setelah menggunakan algoritma ini, proses penjadwalan kuliah lebih cepat, dan tepat.

2. Jurnal : Yana Maulana, 2005, Penjadwalan Perkuliahan Dengan Menggunakan Algoritma Max-Min Ant System

(Maulana, 2005) Dalam penelitiannya menyebutkan bahwa algoritma *Max-Min Ant System* dapat diterapkan dalam pemecahan masalah penjadwalan kuliah, dengan hasil yang baik (tidak terjadi bentrok antar komponen-komponen yang

dijadwalkan). Namun waktu proses penjadwalan dengan algoritma *Max-Min Ant System* sangat dipengaruhi oleh jumlah komponen yang dijadwalkan, semakin besar kasus maka waktu yang diperlukan akan semakin lama pula

3. Jurnal : Eko Zulkaryanto dan Musthofa, 2011, Sistem Penjadwalan Ujian Menggunakan *Answer Set Programming*

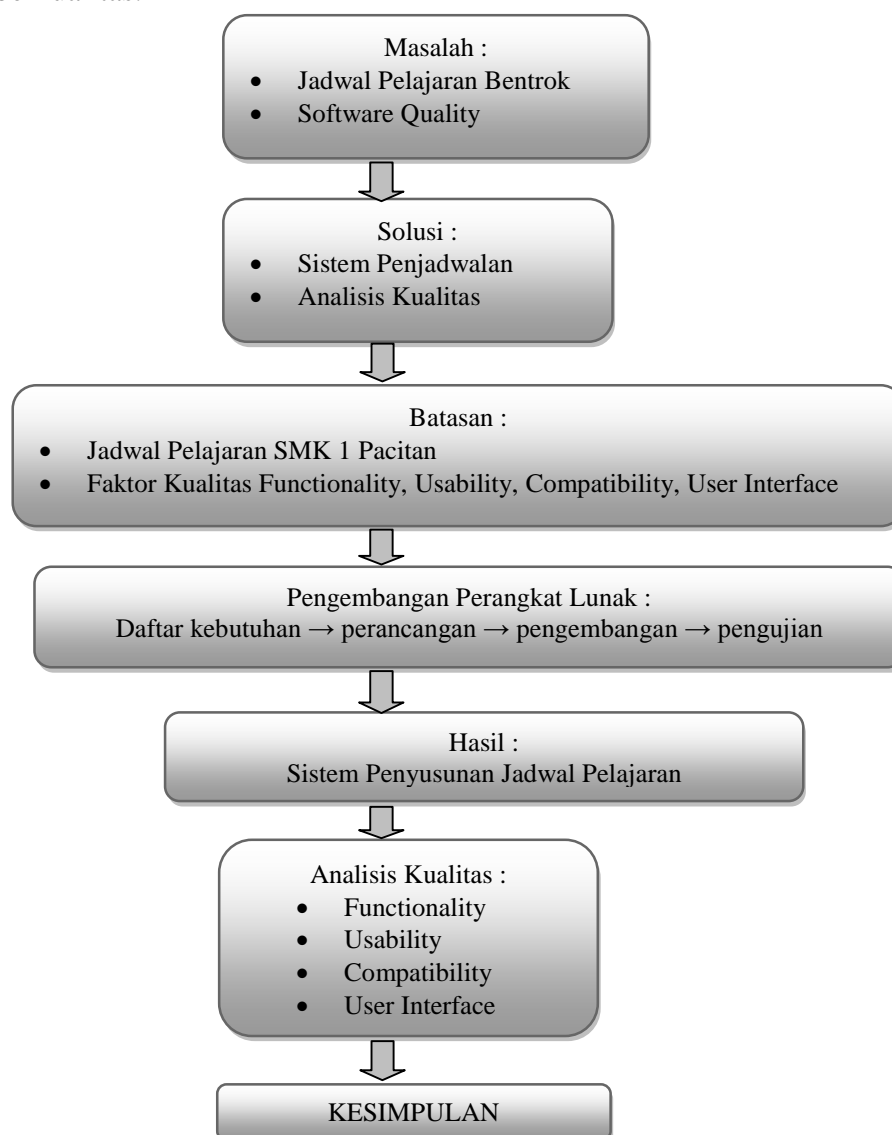
(Eko & Mushtofa, 2011) Dalam menyebutkan bahwa ASP telah mampu memodelkan penyelesaian permasalahan penjadwalan ujian Program Studi S1 Sistem Mayor-Minor Institut Pertanian Bogor. Pemodelan penyelesaian permasalahan penjadwalan berhasil untuk *dataset* yang dibuat per Fakultas dan belum berhasil dilakukan untuk data keseluruhan.

N. Kerangka Berfikir

Dunia pendidikan yang kian berkembang menuntut pihak kurikulum sekolah untuk mampu menyusun jadwal pelajaran sedemikian hingga tidak ada bentrokan antara guru, kelas, pelajaran serta ruang kelas yang digunakan. Maka dibutuhkan suatu perangkat lunak yang dapat membantu pekerjaan tersebut secara efisien.

Perangkat lunak yang dibutuhkan bukan berarti dapat menggantikan peran penjadwalan secara penuh, namun setidaknya kegiatan penjadwalan tersebut dapat diselesaikan secara tepat dan dapat mengurangi berbagai beban permasalahan yang dihadapi ketika dilakukan dengan metode konvensional. Perangkat lunak yang sampai sekarang digunakan masih memiliki beberapa kelemahan, baik dari usability, kompatibilitas, antarmuka pengguna, maupun fungsionalitasnya. Sehingga ini menjadikan suatu ide untuk mengembangkan perangkat lunak baru

yang lebih inovatif mengadopsi dari system yang sudah ada kemudian digabungkan dengan teknik-teknik baru agar lebih bernilai bagi penggunanya, khususnya pihak kurikulum sekolah. Untuk mengembangkan perangkat lunak tersebut, maka selain dibutuhkan ketrampilan rekayasa juga diperlukan pengujian perangkat lunak tersebut dari beberapa faktor seperti usability, functionality, compatibility dan user interface. Sehingga perangkat lunak yang dihasilkan akan lebih berkualitas.



Gambar 7 : Kerangka Berfikir

O. Pertanyaan Peneliti

Berdasarkan kerangka berpikir yang telah dijelaskan sebelumnya, Penulis merumuskan beberapa pertanyaan penelitian antara lain :

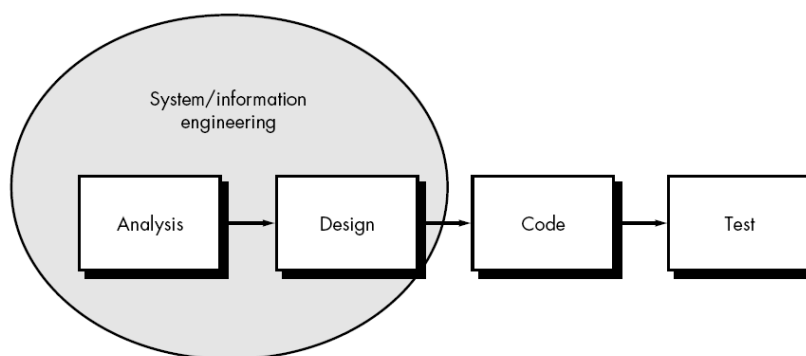
1. Apakah Aplikasi Sistem Penyusunan Jadwal Pelajaran Sekolah yang dikembangkan dalam penelitian ini memenuhi standar faktor kualitas *functionality*?
2. Apakah Aplikasi Sistem Penyusunan Jadwal Pelajaran Sekolah yang dikembangkan dalam penelitian ini memenuhi standar faktor kualitas *usability*?
3. Apakah Aplikasi Sistem Penyusunan Jadwal Pelajaran Sekolah yang dikembangkan dalam penelitian ini memenuhi standar faktor kualitas *compatibility*?
4. Apakah Aplikasi Sistem Penyusunan Jadwal Pelajaran Sekolah yang dikembangkan dalam penelitian ini memenuhi standar faktor kualitas *user interface*?

BAB III

METODE PENELITIAN

A. Prosedur Pengembangan.

Penelitian ini menggunakan metode pengembangan perangkat lunak *classic life cycle* atau model air terjun yang dimulai dari tahap analisis, perancangan, pengembangan dan pengujian (Pressman,2001:29).



Gambar 8 : Classic Life Cicle

1. Tahap Analisis.

Tujuan tahap ini adalah menentukan persyaratan perangkat lunak yang dibuat agar nantinya dapat bermanfaat bagi pihak yang menggunakannya. Persyaratan tersebut meliputi aspek fungsionalitas, usabilitas, serta kompatibilitas.

2. Tahap Perancangan.

Tujuan tahap ini adalah menyiapkan prototipe perangkat lunak. Beberapa kegiatan yang dilakukan seperti mengkaji format laporan yang sudah ada, prosedur-prosedur kerja, serta dokumentasi lain yang kemudian digunakan untuk mendukung perangkat lunak yang dibuat agar nantinya sesuai dan bermanfaat bagi penggunanya.

3. Tahap Pengembangan.

Tujuan tahap ini adalah untuk menghasilkan perangkat lunak yang sudah direvisi berdasarkan masukan dari pakar. Tahap ini meliputi: (a) validasi perangkat oleh para pakar diikuti dengan revisi, (b) simulasi yaitu kegiatan mengoperasionalkan perangkat lunak, dan (c) uji coba terbatas dengan kasus yang sesungguhnya. Hasil tahap (b) dan (c) digunakan sebagai dasar revisi. Langkah berikutnya adalah uji coba lebih lanjut dengan siswa yang sesuai dengan kasus sesungguhnya.

4. Tahap pengujian.

Pada tahap ini merupakan tahap penggunaan perangkat lunak yang telah dikembangkan pada skala yang lebih luas misalnya 1 sekolah, di sekolah lain, oleh pengguna yang lain. Hasil pengujian kemudian dianalisis apakah sudah memenuhi standar kualitas perangkat lunak meliputi factor *usability, functionality, compatibility*, dan *user interface*.

B. Desain Penelitian.

Tujuan penelitian ini adalah mengembangkan perangkat lunak Sistem Penyusunan Jadwal Pelajaran Berbasis Web sekaligus memaparkan hasil analisis kualitasnya. Karena tujuannya adalah menghasilkan produk tertentu dan menguji keefektifan produk tersebut maka metode penelitian yang digunakan adalah *research and development* (Sugiyono,2010:494). Setelah pengembangan perangkat lunak kemudian perangkat lunak divalidasi kepada dosen ahli media. Adapun hasil dari validasi perangkat lunak yang menjelaskan bahwa perangkat

lunak dapat dilakukan pengujian lebih lanjut dengan melakukan beberapa perbaikan yang tidak fatal. Perbaikan tersebut antara lain :

1. Penambahan hak akses pengguna sistem dan perbaikan beberapa kata yang kurang tepat pada label tombol disampaikan oleh Drs. Herman Dwi Surjono, M.Sc, M.T, Ph.D
2. Penambahan petunjuk penggunaan sistem dalam bentuk gambar dan perbaikan beberapa menu yang kurang tepat disampaikan oleh Dr. Drs. Eko Marpanaji, M.T
3. Perbaikan penggunaan bahasa disampaikan oleh Yuniar Indrihapsari, S.T, M.Eng

Setelah dilakukan validasi perangkat lunak beserta perbaikan yang disarankan oleh validator selanjutnya dilakukan pengujian perangkat lunak untuk masing-masing faktor kualitas. Pengujian untuk faktor *functionality*, *compatibility*, dan *user interface* menggunakan metode *black box testing* dengan *test case* yang disiapkan. Sedangkan pengujian faktor *usability* perangkat lunak ini menggunakan metode *surve* dengan kuesioner yang dilaksanakan pada responden sebanyak 30 orang terdiri dari staff kurikulum, guru mata pelajaran dan wali kelas di SMKN 1 Pacitan.

C. Definisi Operasional Variabel Penelitian

Variabel penelitian pada dasarnya adalah segala sesuatu yang bentuk apa saja yang ditetapkan oleh peneliti untuk dipelajari sehingga diperoleh informasi tentang hal tersebut kemudian ditarik kesimpulannya (Sugiyono,2010:58). Variabel penelitian ini adalah kualitas perangkat lunak Sistem Penyusunan Jadwal Pelajaran

Berbasis Web di SMKN 1 Pacitan. Kualitas tersebut meliputi faktor-faktor diantaranya adalah Usabilitas, Kompatibilitas, Fungsionalitas dan Antarmuka Pengguna. Untuk memaparkan kualitas ini dengan cara menggunakan kuesioner. Kualitas diartikan sebagai skor yang diperoleh dari alat pengumpulan data kuesioner.

D. Instrumen dan Teknik Pengumpulan Data

1. Instrumen penelitian.

Instrumen penelitian adalah suatu alat yang digunakan untuk mengukur fenomena alam maupun sosial yang diamati (Sugiyono,2010:146). Instrumen dalam penelitian ini menggunakan *test case* dan kuesioner. *Test case* digunakan untuk menguji faktor *functionality* dan *compatibility*. Kuesioner digunakan untuk menguji faktor *usability* dan memperkuat hasil pengujian faktor *functionality*, *user interface* dan *compatibility*.

Titik tolak dalam menyusun instrument adalah menentukan variable penelitian. Dari variable tersebut diberikan definisi operasionalnya dan selanjutnya ditentukan indikator yang akan diukur. Dari indikator ini kemudian dijabarkan menjadi butir-butir pertanyaan atau pernyataan (Sugiyono,2010:147). Untuk memudahkan penyusunan instrument maka digunakan kisi-kisi instrument.

Tabel 3 : Kisi-kisi kuesioner kualitas perangkat lunak berbasis web.

Variabel	Indikator	No butir	Jumlah
1. Antarmuka Pengguna	<ul style="list-style-type: none"> • Warna • Link • Huruf • Ikon • Tata letak • Bahasa 	1, 2, 3, 4, 5, 6, 7, 8, 9, 10*, 11, 12, 13, 14, 15, 16	16
2. Fungsionalitas	<ul style="list-style-type: none"> • Link • Form • Validasi • Scrollbar • Pemberitahuan 	17, 18, 19, 20, 21, 22, 23	7
3. Kompatibilitas	<ul style="list-style-type: none"> • Browser • Printer • Resolusi layar 	24, 25, 26, 27, 28, 29	6
4. Usabilitas	<ul style="list-style-type: none"> • Kemudahan • Kesederhanaan • Kecepatan • Ketepatan • Kenyamanan • Kejelasan • Keefektifan 	30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48	19
4	22		48

Keterangan : * (pernyataan negatif)

Selanjutnya, instrumen diteliti oleh ahli (*expert judgment*), yaitu oleh 3 dosen antara lain Dr. Drs. Eko Marpanaji, M.T, Drs. Herman Dwi Surjono, M.Sc, M.T, Ph.D dan Handaru Jati, S.T, M.M, M.T, Ph.D yang masing-masing adalah dosen teknik informatika UNY.

Selain kuesioner, instrument yang digunakan berupa *test case* untuk menguji faktor *functionality*, *user interface* dan *compatibility*, secara mendalam. Setiap fungsi yang disediakan dalam sistem ini diuji satu persatu menggunakan *test case*. Tujuannya adalah mengetahui apakah terdapat fungsi yang tidak berjalan. Sistem ini juga dijalankan pada berbagai browser, resolusi layar, jenis printer dan kondisi lain untuk menguji apakah sistem dapat berjalan dengan baik pada setiap kondisi yang diujikan. Faktor *user interface* diuji dengan mengamati setiap komponen *user interface* sesuai *test case*.

2. Uji coba instrumen

Instrumen penelitian yang telah disusun kemudian diujicobakan di luar sampel penelitian. Uji coba instrumen yang berupa kuesioner dilaksanakan di SMKN 5 Yogyakarta dengan jumlah responden 10 orang. Uji coba ini dilakukan untuk mengetahui validitas dan reliabilitas instrumen penelitian yang akan dipergunakan untuk mengungkap kualitas perangkat lunak Sistem Penyusunan Jadwal Pelajaran Berbasis Web.

a. Uji validitas

Validitas diartikan suatu ukuran yang menunjukkan tingkat-tingkat keandalan atau kesahihan instrumen. Uji validitas penelitian ini menggunakan

rumus korelasi total (Sugiyono, 2007:356). Langkah – langkah pokok dalam analisis kesahihan butir pada dasarnya adalah :

1. Menghitung skor faktor dari skor butir.
2. Menghitung korelasi momen tangkar antara butir dengan faktor
3. Mengoreksi korelasi momen tangkar menjadi korelasi bagian total
4. Menguji signifikansi korelasi bagian total itu
5. Menggugurkan butir-butir yang tidak sah.

$$r_{xy} = \frac{n\sum x_1y_1 - (\sum x_1)(\sum y_1)}{\sqrt{\{n\sum x_1^2 - (\sum x_1)^2\}\{n\sum y_1^2 - (\sum y_1)^2\}}}$$

Keterangan:

r_{xy} = korelasi momen tangkar

n = cacah subjek uji coba

$\sum x$ = jumlah skor butir

$\sum x^2$ = jumlah skor butir kuadrat

$\sum y$ = jumlah skor total butir

$\sum y^2$ = jumlah skor total butir kuadrat

$\sum xy$ = sigma tangkar (perkalian skor butir dengan skor total butir)

Uji validitas dengan langkah-langkah tersebut di atas dilakukan dengan menggunakan bantuan software SPSS 15.

b. Uji reliabilitas

Instrument yang reliable berarti instrument yang bila digunakan beberapa kali untuk mengukur obyek yang sama, akan menghasilkan data yang sama (Sugiyono, 2007: 348). Untuk menguji tingkat keterandalan instrumen digunakan

rumus alpha. Adapun rumus tersebut adalah sebagai berikut : (dalam Sugiyono, 2007 : 365)

$$r_{11} = \left[\frac{k}{k-1} \right] \left[1 - \frac{\sum \sigma^2 b}{\sigma^2 t} \right]$$

Keterangan :

r_{11} = reliabilitas instrumen

k = banyaknya butir pertanyaan

$\sum \sigma^2 b$ = jumlah varian butir

$\sigma^2 t$ = varians total

hasil perhitungan menggunakan rumus diatas diinterpretasikan dengan tingkat keandalan koefisien korelasi sebagai berikut :

antara 0,800 s.d. 1,000 adalah sangat tinggi

antara 0,600 s.d. 0,799 adalah tinggi

antara 0,400 s.d. 0,599 adalah cukup

antara 0,200 s.d. 0,399 adalah rendah

antara 0,000 s.d. 0,199 adalah sangat rendah

3. Teknik Pengumpulan Data

Teknik pengumpulan data penelitian ini menggunakan kuesioner tertutup , observasi dan dokumentasi.

E. Teknik Analisis Data

Dalam penelitian ini, setelah data penelitian diperoleh maka untuk menganalisisnya terdapat beberapa cara. Untuk analisis faktor *functionality* dilakukan dengan meneliti setiap fungsi yang diuji berdasarkan *test case*. Jumlah total fungsi yang diujikan dikurangi dengan fungsi yang gagal saat diuji kemudian

diubah ke bentuk persen. Misalnya jika hanya 10 fungsi yang berjalan dari 20 fungsi yang diujikan, maka nilai kualitas faktor *functionality* adalah 50%. Perhitungannya adalah $10:20 \times 100\% = 50\%$.

Analisis faktor *compatibility* didapatkan dengan mengamati satu-persatu sistem ini pada berbagai jenis perangkat yang diujikan sesuai *test case*. Terdapat 4 hal penting dalam pengujian *compatibility* untuk aplikasi berbasis web yaitu *browser compatibility*, *operating system compatibility*, *mobile browsing* dan *printing options* (<http://www.softwaretestinghelp.com/web-application-testing>). Ada 1 hal yang tidak diujikan pada sistem ini yaitu *mobile browsing* karena sistem ini membutuhkan dimensi layar yang besar minimal 800 x 600px dan penggunaannya pun tidak pada *mobile platform*. Hasil pengujian dari setiap *test case* selanjutnya didokumentasikan untuk memudahkan pengamatan.

Analisis faktor *user interface* dihitung dengan membandingkan setiap komponen *user interface* dengan standar *user interface* untuk aplikasi berbasis web. Berikut standar *user interface* untuk aplikasi berbasis web yang digunakan.

Tabel 4 : Standar pengembangan *User Interface* aplikasi berbasis web

Component	Description
Screen Font Type	Ensure that the screen font family matches from screen to screen. Mismatching fonts within the same sentence and overuse of different fonts can detract from the professionalism of your software user interface.
Screen Font Sizes	Ensure that the screen font sizes match from screen to screen. A good user interface will have an accompanying style guide that explicitly defines the font type and size for headers, body text, footers, etc.
Colors	Ensure that screens do not use different color sets as to cause an inconsistent and poorly thought
Icons	Ensure that icons are consistent throughout your application by using a common icon set. For example, a BACK link that contains an icon next to it should not have a different icon on one screen versus another. Avoid free clip

Narrative Text	Having narrative text (screen instructions) is a great way to communicate how to use a specific screen. Ensure that narrative text appears at the same location on the screen on all screens.
Brevity	Ensure that narrative text, error messages and other instructions are presented in laymen's terms but are brief and to
Dialog Box Consistency	Use a style guide to document what choices are available for dialog boxes. You should not have Save/Cancel dialog on one screen and an OK/Cancel on another, this is inconsistent.
Links	If your application has links on the screen (e.g. Save as Spreadsheet, Export, Print, Email, etc.), ensure that the links have consistent spacing between them and other links, that the links appear in the same order from screen to screen, and that the color of the links are consistent.
Menus	If your application has menu items, ensure that menu items that are not applicable for the specific screen are disabled and the order in which each menu item appears is consistent from screen to screen.
Buttons	If your application has buttons (e.g. Submit, OK, Cancel, etc), ensure that the buttons appear in a consistent order from screen to screen (e.g. Submit then Cancel).
Abbreviation Inconsistencies	If your screens contain abbreviations (e.g. Nbr for number, Amt for amount, etc), the abbreviations should be consistent for all screens in your application. Again, the style guide is key for ensuring this.
Delete Confirmations	It is a good practice to ask the user to confirm before deleting an item. Create test cases to ensure that all delete operations require the confirmation. Taking this a step further, it would also be great to allow clients to turn off specific confirmations if they decide to do this.
Save Confirmations	It is good practice to ask the user to confirm an update if updates are made and they navigate to another item before explicitly saving. Create test cases to ensure that all record movement operations require the confirmation when updates are made. Taking this a step further, it would also be great to allow clients to turn off specific confirmations if they decide to do this.
Grammar and Spelling	Ensure that you have test cases that look for grammar or spelling errors.
Shortcuts	If your application allows short cut keys (like CTRL+S to save), ensure that all screens allow using of the consistent shortcuts.

Sumber : <http://blog.smartbear.com>

Hasil pengujian berupa kuesioner menggunakan analisis deskriptif kuantitatif. Skor untuk pernyataan dibagi menjadi dua bagian, yaitu skor pernyataan positif dan skor pernyataan negatif. Adapun skor jawaban sebagai berikut :

Tabel 5 : Skor Instrumen

Kode	Keterangan	Pernyataan negatif	Pernyataan positif
		Skor	Skor
1	Tidak setuju	4	1
2	Kurang setuju	3	2
3	Setuju	2	3
4	Sangat setuju	1	4

Data dari responden kemudian dijumlahkan berdasarkan kategori faktornya dan dijumlahkan nilai total semua faktor, sehingga diperoleh jumlah nilai dari masing-masing faktor dan total nilainya. Untuk menganalisisnya dicari nilai maksimal dan nilai minimal. Nilai maksimal diperoleh dengan asumsi semua jawaban bernilai 4 untuk 30 responden. Maka nilai maksimalnya adalah 120 untuk setiap butir pertanyaan. Nilai minimal diperoleh dengan asumsi semua jawaban bernilai 1 dari 30 responden, maka nilai minimalnya adalah 30 untuk setiap butir pertanyaan. Sedangkan nilai maksimal setiap faktor diperoleh dengan mengalikan jumlah pertanyaan (n) dengan nilai maksimal per item (120). Berikut rincian skor maksimal dan minimal untuk setiap faktor.

Tabel 6 : Rincian skor maksimal minimal kuesioner

No	Faktor	Jumlah pertanyaan	nilai minimal	Nilai maksimal	Range
1	Functionality	7	7	28	21
2	User interface	16	16	64	48
3	Compatibility	6	6	24	18
4	Usability	19	19	76	57
Total		48	48	192	144

Dari data tersebut, kemudian dapat disusun kategori penilaian kuesioner menjadi 5 kelas. Rentang nilai untuk masing-masing kelas didapatkan dari range dibagi 5. Maka kelas penilaian untuk setiap faktor dan keseluruhan adalah sebagai berikut.

Tabel 7 : Pengkategorian Skor kuesioner

Kategori Nilai	Rentang nilai per item pertanyaan per faktor			
	<i>Functionality</i>	<i>User interface</i>	<i>Compatibility</i>	<i>Usability</i>
Sangat baik	23,8 - 28	54,4 – 64	20,4 - 24	64,6 - 76
Baik	19,6 - 23,8	44,8 - 54,4	16,8 - 20,4	53,2 - 64,6
Cukup	15,4 - 23,8	35,2 - 44,8	13,2 - 16,8	41,8 - 53,2
Kurang	11,2 - 15,4	25,6 - 35,2	9,6 - 13,2	30,4 - 41,8
Sangat kurang	7 - 11,2	16 - 25,6	6 - 9,6	19 - 30,4

Untuk 30 responden maka tabel pengkategorian nilai diperoleh sebagai berikut.

Tabel 8 : Pengkategorian Skor total kuesioner

Kategori Nilai	Rentang nilai total per faktor			
	<i>Functionality</i>	<i>User interface</i>	<i>Compatibility</i>	<i>Usability</i>
Sangat baik	714-840	1632-1920	612-720	1938-2280
Baik	588-714	1344-1632	504-612	1596-1938
Cukup	462-588	1056-1344	396-504	1254-1596
Kurang	336-462	768-1056	288-396	912-1254
Sangat kurang	210-336	480-768	180-288	570-912

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

A. Pengembangan Perangkat Lunak

1. Analisis Kebutuhan

Berdasarkan dari hasil observasi peneliti ke sekolah yang bersangkutan yaitu SMKN 1 Pacitan, jadwal pelajaran setiap awal semester disusun oleh staff kurikulum bekerja sama dengan beberapa guru mata pelajaran dan wali kelas. Hasil dari kerjasama tersebut yaitu berupa kontrak mengajar yang isinya adalah pembagian jumlah jam mengajar untuk masing-masing guru sesuai dengan bidang kompetensinya, dan rata-rata untuk guru normatif dan adaptif mendapat jumlah jam mengajar antara 18 sampai dengan 24 jam pelajaran untuk setiap minggu. Kontrak mengajar tersebut selanjutnya dibagi untuk masing – masing kelas guna memenuhi kebutuhan jam pelajaran yang telah disetujui oleh kepala sekolah. Daftar kebutuhan KBM untuk setiap jurusan ini biasa disebut dengan struktur kurikulum.

Di dalam struktur kurikulum dijelaskan kebutuhan jumlah jam pelajaran dari masing-masing mata pelajaran. Sebagai contoh untuk mata pelajaran Pendidikan Agama yang merupakan mata pelajaran normatif dibutuhkan 2 jam pelajaran setiap minggu untuk kelas X serta dibutuhkan 3 jam pelajaran untuk kelas XI dan kelas XII. Sedangkan untuk mata pelajaran Bahasa Inggris sebagai mata pelajaran adaptif dibutuhkan 5 jam pelajaran setiap minggu untuk kelas X dan kelas XI

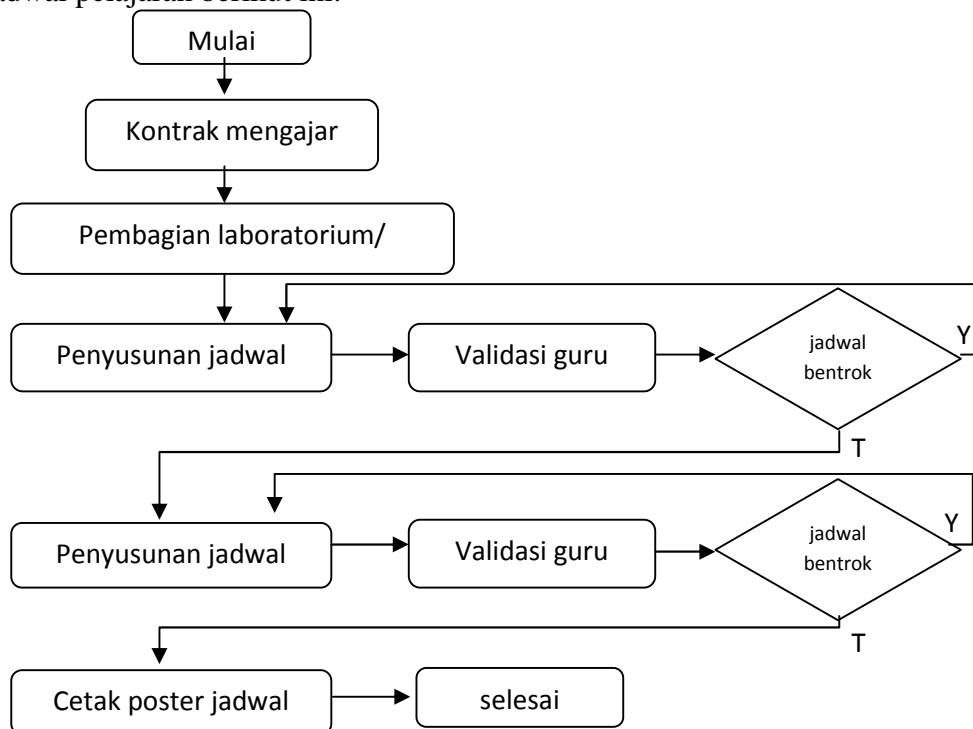
serta dibutuhkan 4 jam pelajaran untuk kelas XII. Rincian Struktur kurikulum yang lebih lengkap terdapat di halaman lampiran penelitian ini.

Setelah pembagian kontrak mengajar dari masing-masing guru sudah mencukupi, langkah selanjutnya adalah pembagian laboratorium/ bengkel untuk mata pelajaran jenis praktikum. Hal ini dilakukan untuk menghitung kebutuhan jumlah lab/bengkel untuk memenuhi kebutuhan KBM dari semua kelas. Di SMKN 1 Pacitan terdapat 10 laboratorium komputer yang dibagi menjadi 4 bagian, yaitu 2 lab untuk jurusan Multimedia, 2 lab untuk jurusan RPL, 2 lab untuk TKJ, 2 lab untuk TGB dan 2 lab untuk mata pelajaran KKPI. Berbeda dengan bidang kriya dan busana yang hanya memiliki 2 lab saja yang sudah mencukupi untuk semua kelas, ini disebabkan karena penggunaan lab komputer lebih dari 1 kelas dalam 1 lab secara bersamaan tidak memungkinkan, sedangkan penggunaan bengkel praktek untuk kriya dan busana lebih dari 2 kelas secara bersamaan masih memungkinkan.

Setelah pembagian lab/bengkel sudah mencukupi kemudian penyusunan jadwal pelajaran dilakukan sesuai dengan plot masing-masing guru. Menurut nara sumber, jadwal pelajaran disusun dengan mengutamakan jam praktikum terlebih dahulu, karena jam praktikum membutuhkan lebih banyak jam pertemuan, selain itu penggunaan sarana praktikum juga harus dibagi agar mencukupi kelas yang melakukan praktik. Penyusunan jadwal pelajaran dilakukan dengan mengisikan pertemuan untuk setiap guru pada hari awal (senin) kemudian jika hari senin sudah penuh berpindah ke hari selasa dan seterusnya. Ini dilakukan terus hingga seluruh guru mendapatkan jadwal sesuai kontrak mengajarnya. Jika terdapat

bentrok laboratorium / bengkel, maka salah satu kelas digeser hari praktikumnya.

Setelah jadwal pelajaran disusun, maka langkah selanjutnya adalah validasi kepada masing-masing guru, yaitu dengan mencetak jadwal mengajar masing-masing guru dan membagikan kepada guru yang bersangkutan. Masalah yang sering terjadi disini adalah terdapat kekurangan jam mengajar atau justru malah kelebihan jam mengajar setiap minggunya. Sehingga harus dilakukan perbaikan jadwal pelajaran lagi. Jika semua jadwal mengajar masing-masing guru sudah benar, maka langkah terakhir adalah mencetak jadwal pelajaran model poster yang ditempel pada papan pengumuman sekaligus sebagai laporan kepada kepala sekolah dan mencetak jadwal pelajaran untuk masing-masing kelas untuk dibagikan kepada wali kelas. Sampai disini penyusunan jadwal pelajaran dianggap selesai. Dari uraian di atas, maka dapat bagan langkah-langkah dalam penyusunan jadwal pelajaran berikut ini.



Gambar 9 : Bagan alur penggunaan perangkat lunak

Bagan di atas selanjutnya dijadikan dasar untuk merancang sistem penyusunan jadwal pelajaran berbasis web dalam penelitian ini.

Adapun kebutuhan dari sistem yang dirancang antara lain :

- a. Sistem memiliki pembagian hak akses yaitu untuk administrator, operator, guru, dan siswa.
- b. Sistem dapat mencetak jadwal pelajaran untuk guru, kelas, dan untuk poster.
- c. Untuk keperluan entri data dapat dilakukan secara manual atau secara masal (import data).
- d. Terdapat fitur untuk penyusunan jadwal secara otomatis maupun manual.
- e. Sistem dapat menyimpan hasil kerja dalam format dokumen.

Sesuai kebutuhan sistem tersebut, maka penggunaan algoritma untuk penyusunan jadwal secara otomatis juga disesuaikan, komponen yang dijadwalkan adalah jam pelajaran, guru, mata pelajaran dan kelas. Sedangkan *constraint* yang diterapkan antara lain :

- a. Tidak ada bentrok pertemuan antara satu guru dengan guru lain pada kelas dan waktu yang sama.
- b. Satu guru tidak mengajar pada 2 kelas yang berbeda atau lebih pada waktu yang sama.
- c. Satu guru dapat mengampu 2 mata pelajaran atau lebih.
- d. Setiap pertemuan dapat menentukan batas akhir jam pertemuan.

Setiap *constraint* tersebut harus dipenuhi dalam menyusun jadwal pelajaran. adapun algoritma yang diterapkan adalah *blind search*. Sehingga langkah penyelesaiannya adalah sebagai berikut.

Algoritma Blind Search

```
input hari, jampel, kelas
input pertemuan[n]
set berhasil = 0
set gagal = 0
create array jadwal[hari][kelas][jampel]
while count(pertemuan) > 0
    set posisi = 0
    set selesai = false
    while posisi < count(jadwal) AND selesai=false
        if jadwal(posisi) = NULL then
            if cek_constraint = true then
                alokasikan pertemuan[n] = jadwal[posisi]
                set selesai = true
                count(pertemuan)= -1
                set berhasil = +1
                n = +1
            end if
        end if
        set posisi= +1
        if posisi = count(jadwal) AND cek_constraint = false then
            set selesai = true
            set gagal = +1
            count(pertemuan)= -1
            n = +1
        end if
    end while
end while

write "[berhasil] pertemuan berhasil dialokasikan"
write "[gagal] pertemuan gagaldialokasikan"
write [jadwal]
```

Penjelasan dari algoritma tersebut adalah sebagai berikut :

1. Masukkan data pokok untuk menyusun matriks jadwal pelajaran yaitu data hari, data jam pelajaran, data kelas
2. Ciptakan array kosong dari data pokok yang sudah ditetapkan.
3. Tutup array yang digunakan untuk jadwal statik seperti upacara, istirahat, dll.

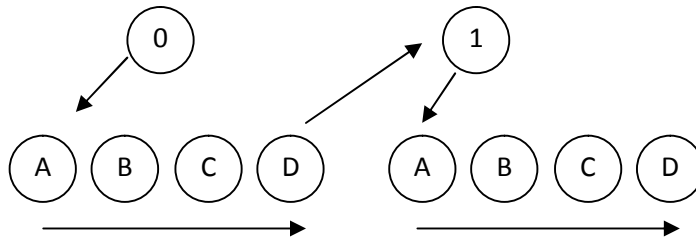
Sampai pada langkah ini struktur array dapat dilihat pada gambar berikut :

0	1	1	1	1
1	1	1	2	0
2	1	1	3	0
...
10	2	1	1	0
11	2	2	1	0
12	2	2	2	0
13	2	2	2	0
..
...
50	6	6	1	0
↑	↑	↑	↑	↑
Indek	hari	kelas	waktu	Isi/kosong

Gambar 10 : Struktur data jadwal pelajaran

4. Tambahkan pertemuan satu per satu sesuai kontrak mengajar guru beserta *constraint* batas akhir pertemuan dan panjang pertemuan pada array pertemuan untuk dijadwalkan.
5. Setiap pertemuan dialokasikan pada matriks jadwal mulai dari array 0 yaitu hari senin jam pertama berikut sampai array terakhir yaitu hari sabtu jam ke terakhir, jika melanggar salah satu *constraint* maka berpindah ke array berikutnya hingga menemukan posisi yang tidak melanggar *constraint*.
6. Jika sampai pada array terakhir tidak menemukan posisi yang cocok, maka pertemuan dinyatakan gagal dialokasikan dan memulai dari array pertama untuk pertemuan berikutnya.
7. Proses diulangi hingga setiap pertemuan dinyatakan berhasil atau gagal dialokasikan pada matriks jadwal.

Ilustrasi pencarian posisi pertemuan pada langkah ini dapat dilihat pada gambar berikut ini.

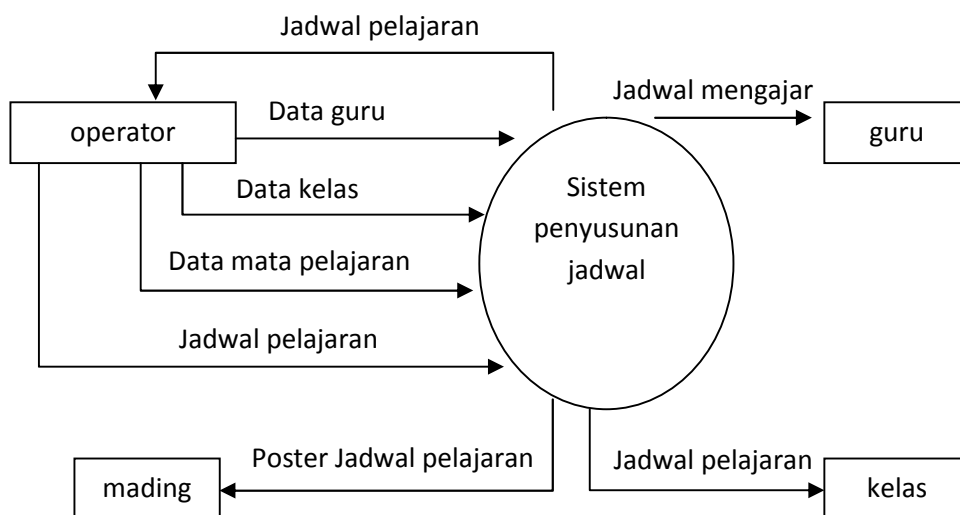


Gambar 11: Proses pencarian dalam penyelesaian jadwal pelajaran

Pada gambar tersebut dapat dilihat proses pencarian posisi jadwal dimulai dari node 0 yaitu hari senin pada sub-node A yaitu jam pertama sampai jam terakhir. Selanjutnya ke hari selasa jam pertama sampai terakhir. Maka dengan penerapan algoritma *blind search* ini diharapkan setiap pertemuan dapat dialokasikan pada matriks jadwal pelajaran dan tidak melanggar setiap *constraint* yang ditetapkan.

2. Perancangan sistem

Setelah mendapatkan kebutuhan dari sistem yang akan dirancang, selanjutnya adalah pembuatan proses bisnis yang pada penelitian ini digambarkan dengan DFD (*data flow diagram*). Berikut proses bisnis dari sistem yang akan dirancang:



Gambar 12 : DFD perangkat lunak

Dari bagan di atas terlihat bahwa sebagian besar pengguna dari sistem adalah operator. Sedangkan guru dan kelas / siswa hanya mendapatkan hak akses baca (*read only*). Oleh sebab itu setelah sistem selesai dibuat, kebanyakan validasi dan verifikasi lebih banyak dilakukan dengan operator. Pembagian hak akses sistem lebih jelasnya dapat dilihat pada table berikut.

Tabel 9 : Pembagian Hak Akses perangkat lunak

No	Menu	Submenu	Penjelasan	Hak akses
1	Admin	Pengguna sistem	Menampilkan, mengubah, menambah dan menghapus pengguna sistem beserta hak aksesnya	Administrator
2	Setting	Identitas sekolah	Mengatur informasi identitas sekolah yang bersangkutan yaitu nama sekolah, alamat, tahun ajaran dan semester	Administrator, operator
3	Setting	Jam pelajaran	Mengatur jam pelajaran sekolah yang bersangkutan	Administrator, operator
4	Setting	Hari KBM	Mengatur hari efektif KBM sekolah yang bersangkutan	Administrator, operator
5	Entri	Kelas	Menampilkan, menambah, mengubah dan menghapus data kelas. Atribut kelas terdiri dari nama kelas dan tingkat kelas.	Administrator, operator
6	Entri	Guru	Menampilkan, menambah, mengubah dan menghapus data guru pengajar untuk jadwal pelajaran. atribut	Administrator, operator

			guru terdiri dari nama guru, NIP, nama singkat dan jumlah jam mengajar.	
7	Entri	Mata pelajaran	Menampilkan, menambah, mengubah dan menghapus data mata pelajaran untuk jadwal pelajaran	Administrator, operator
8	Entri	Kode pelajaran	Menampilkan, menambah, mengubah dan menghapus data kode pelajaran untuk jadwal pelajaran, yaitu pasangan antara guru dengan mata pelajaran yang diampu beserta kode pelajaran dan warna.	Administrator, operator
9	Jadwal	Jadwal tetap	Menampilkan, menambah, mengubah dan menghapus data jadwal tetap seperti jam istirahat, upacara, dsb.	Administrator, operator
10	Jadwal	Jadwal kelas	Menampilkan, menambah, mengubah dan menghapus antrian data pertemuan kelas yang akan dialokasikan secara otomatis oleh sistem.	Administrator, operator
11	Jadwal	Auto generate	Mengalokasikan pertemuan kelas yang telah diantrikan paka jadwal kelas ke jadwal utama secara otomatis.	Administrator, operator
12	Jadwal	Jadwal	Menampilkan, menambah,	Administrator,

		utama	mengubah dan menghapus jadwal pelajaran utama	operator
13	Jadwal	Rekap per guru	Menampilkan jumlah pertemuan yang sudah dialokasikan ke jadwal utama untuk masing-masing guru	Administrator, operator
14	Cetak	Cetak jadwal per kelas	Menampilkan dan mengunduh file cetak jadwal pelajaran per kelas.	Administrator, operator, guru, guest
15	Cetak	Cetak Jadwal per guru	Menampilkan dan mengunduh file cetak jadwal pelajaran per guru.	Administrator, guru, operator
16	Cetak	Cetak semua jadwal	Menampilkan dan mengunduh file cetak jadwal pelajaran utama.	Administrator, operator, guru

Berdasarkan tabel di atas, secara umum sistem yang dirancang akan memiliki 7 halaman utama yaitu :

- a. Halaman login
- b. Halaman awal (*home*)
- c. Halaman admin
- d. Halaman pengaturan (*setting*)
- e. Halaman entri data
- f. Halaman entri jadwal
- g. Halaman cetak.

Pada halaman awal juga dirancang untuk menampilkan petunjuk singkat penggunaan sistem agar lebih memudahkan bagi pengguna awam.

3. Pengembangan sistem

Pengembangan sistem dilakukan dengan membuat desain tampilan dari masing-masing komponen, misalnya untuk tampilan form, tabel, menu, serta peletakan tombol dan navigasi. Untuk kemudahan entri data, fitur hapus, edit, pencarian, pengurutan (*sorting*) serta ekspor data diintegrasikan dengan tabel data. Sehingga pada sebuah tabel sudah dapat memuat masing-masing fitur tersebut. Berikut ini adalah contoh dari tampilan tabel data.

::data guru mengajar

Muat Ulang Ekspor Pencarian Show / hide columns

tampilkan 10 baris

nomor	Nama Singkat	Nama Lengkap	NIP	Jumlah Jam
1	1	JUMIRAN, S.Ag	123 445 77 8888	12
2	2	Drs. SUCHARTO		18
3	3	Kl. delete S.Pd		15
4	4	Dr. edit I		18
5	5	UMI KHASANAH, S.Ag		10
6	6	SITI RAHAYU, S.Ag		18
7	7	Drs. HADI WINARNO		16
8	8	Drs. SUTADI		12
9	9	SOEPANDI, S.pd		12
10	11	IMBANG SOURİYATI, S.Pd		16

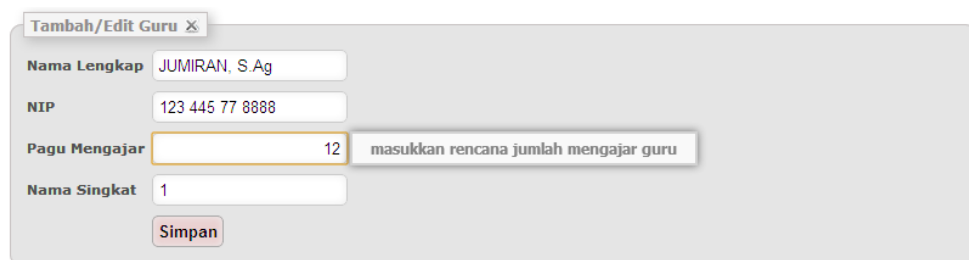
data 1 s.d 10 dari 138 awal Sebelum 1 2 3 4 5 Berikut akhir

Gambar 13 : Tampilan tabel data guru

Pada gambar di atas terlihat bahwa ketika pengguna memilih salah satu data maka muncul menu delete dan edit. Kemudian pada bagian atas (*header*) tabel juga ditambahkan fitur pencarian data, ekspor data dan filter kolom. Untuk pengurutan data (*sorting*) dapat dilakukan dengan klik pada judul kolom, maka tampilan data akan diurutkan secara *ascending* atau *descending*. Pada bagian bawah tabel (*footer*) terlihat adanya fitur halaman yang berfungsi untuk memudahkan navigasi data yang ditampilkan, sehingga jika jumlah data melebihi jumlah baris pada satu halaman, maka data akan ditampilkan pada halaman tabel

berikutnya. Masing – masing fitur tersebut dapat disediakan dengan plugin jquery yang sudah tersedia dan bebas untuk digunakan.

Kemudian untuk tampilan entri data masih menggunakan form seperti halaman web pada umumnya. Juga terdapat pesan bantuan (*tooltips*) pada beberapa field yang membutuhkan keterangan. Berikut contoh tampilan form untuk sistem ini.



The image shows a web form titled "Tambah/Edit Guru" with a close button (X). The form contains the following fields and values:

- Nama Lengkap:** JUMIRAN, S.Ag
- NIP:** 123 445 77 8888
- Pagu Mengajar:** 12. A tooltip is visible next to this field with the text "masukkan rencana jumlah mengajar guru".
- Nama Singkat:** 1

At the bottom of the form is a red button labeled "Simpan".

Gambar 14 : Tampilan form input data guru

Proses validasi data dilakukan dua kali, yaitu ketika pengguna memasukkan data pada field dan ketika pengguna menekan tombol simpan. Validasi dilakukan pada format data dan kebutuhan data, yaitu jika pengguna memasukkan format yang salah pada form dan jika terdapat field yang diperlukan masih kosong.

Kemudian setelah dibuat *template* untuk tabel dan form, berikutnya adalah pembuatan model database. Karena data harus dikelompokkan maka dirancang model database. Pada penelitian ini database menggunakan RDBMS MySQL. Terdapat beberapa tabel untuk memenuhi kebutuhan dari sistem ini. Untuk tabel data awal yaitu tabel guru, tabel kelas, tabel jam pelajaran, tabel hari, tabel identitas sekolah, dan tabel admin. Kemudian dari tabel guru dan mata pelajaran diturunkan pada tabel kode pelajaran, karena terdapat beberapa guru yang mengajar lebih dari 1 mata pelajaran. Untuk penyusunan jadwal pelajaran dari masing-masing tabel tersebut direlasikan ke tabel pertemuan yang pada tabel ini

dipasang constraint agar masing-masing guru dan kelas tidak terjadi bentrok pertemuan, selain itu juga dari tabel pertemuan ini jadwal pelajaran di tampilkan dan dicetak.


Setelah perancangan database, berikutnya adalah pembuatan fitur Create Read Update Delete (CRUD) untuk masing-masing halaman. dimulai dari halaman entri guru, entri kelas, entri mata pelajaran dan seterusnya. Masing-masing halaman tersebut menampilkan data dari tabel database. Setelah semua halaman untuk data pokok selesai, berikutnya adalah pembuatan halaman entri jadwal pelajaran. Pada halaman ini berbeda dengan halaman entri sebelumnya. Tampilan dari tabel jadwal pelajaran hanya menampilkan kolom hari dan jampelajaran secara mendatar dan menampilkan baris kelas secara menurun. Berikut tampilan dari tabel jadwal pelajaran.

Filter Tingkat Kelas ✕

Tingkat

☐ 1
☐ 2
☐ 3
☒ SEMUA

Simpan



Hapus semua

jampel	2	3	4	5	6	7	8	9	2	3	4	5	6	7	8	9	2	3	4	5	6	7	8	9	2	3	4	5	6	7	8	9	2	3	4	5	6	7	8	9
X TGB	FIS 49	FIS 49																																						
X MM2									OR 23								OR 23																							
X MM1	B.ING 47	B.ING 47																																						
X TKJ1																	PAL 1																							
X TKJ2																																								
X TKJ3																																								
X RPL1																																								
X RPL2																																								
X KKKU																																								
X KT																																								
XII TKJ1																																								
XII TKJ2																																								
XII TKJ3																																								
jampel	2	3	4	5	6	7	8	9	2	3	4	5	6	7	8	9	2	3	4	5	6	7	8	9	2	3	4	5	6	7	8	9	2	3	4	5	6	7	8	9
hari	senin								selasa								rabu								kamis								jumat							

Gambar 15 : Tampilan jadwal pelajaran utama

Pada gambar tersebut terlihat jadwal pelajaran untuk masing-masing kelas secara mendatar dan jadwal pelajaran untuk setiap jam pelajaran secara menurun. Ketika pengguna memilih salah satu cell yang sudah isi dengan pertemuan, maka muncul menu edit dan hapus. Di atas tabel juga terdapat menu hapus semua untuk mengosongkan jadwal. Selain itu untuk membatasi kelas yang ditampilkan juga terdapat filter berdasarkan tingkatan kelas.

Bagian terakhir yang dibuat adalah format cetak dari jadwal pelajaran. terdapat tiga jenis format cetak yaitu untuk guru, untuk kelas dan cetak untuk poster masing. Untuk memudahkan pencetakan maka hasil dari masing-masing format tersebut disimpan dalam format file PDF yang selanjutnya dapat dicetak ke printer menggunakan *acrobat reader* atau aplikasi dokumen PDF sejenisnya.

4. Pengujian

a. Pengujian faktor *functionality*.

Pengujian faktor *functionality* dilakukan dengan mencoba satu persatu fungsi utama dalam sistem ini menggunakan *test case*. Berikut contoh hasil pengujian yang telah dilakukan.

Tabel 10 : *Test Case login-logout*

Test case id	Login/logout
Purpose	Menguji fungsi login dan logout
Assumptions	Sistem diakses pada web browser
Test data	<ol style="list-style-type: none"> 1. Gagal login 2. Berhasil login (admin) 3. Berhasil login (operator) 4. Berhasil login (guru) 5. Berhasil logout
Steps	Klik menu login pada halaman utama, <ol style="list-style-type: none"> 1. Masukkan username : coba, password : coba, klik LOGIN 2. Masukkan username : admin, password : rahasia, klik LOGIN 3. Masukkan username : operator, password :12345, klik LOGIN 4. Masukkan username : guru, password : rahasia12345, klik LOGIN 5. Setelah login, Klik menu Keluar

Expected result:	<ol style="list-style-type: none"> 1. Muncul pesan “gagal login” dan tetap pada halaman login 2. Berpindah ke halaman utama dengan menu admin 3. Berpindah ke halaman utama dengan menu operator 4. Berpindah ke halaman utama dengan menu guru 5. Berpindah ke halaman utama dengan menu Guest
Actual result:	<ol style="list-style-type: none"> 1. Muncul pesan “gagal login” dan tetap pada halaman login 2. Berpindah ke halaman utama admin 3. Berpindah ke halaman utama dengan menu operator 4. Berpindah ke halaman utama dengan menu guru 5. Berpindah ke halaman utama dengan menu Guest
Pass/Fail:	Lolos

Hasil pengujian diatas didasarkan pada faktor *functionality*. Pengujian untuk fungsi insert, delete, update, impor, ekspor dan pencarian dilakukan pada setiap halaman entri yaitu entri kelas, guru, mapel dan kode pelajaran dengan metode yang sama hasilnya LOLOS. Hasil pengujian faktor *functionality* selengkapnya dapat dilihat pada halaman lampiran penelitian ini. Berdasarkan hasil pengujian di atas maka sistem dapat berfungsi dengan baik untuk menyusun jadwal pelajaran secara otomatis dan manual.

b. Pengujian faktor *compatibility*.

1) *Browser Compatibility*

Pengujian faktor *compatibility* untuk jenis *web browser* dilakukan dengan mencoba menjalankan sistem pada beberapa web browser yang sering digunakan. Pengujian dianggap lolos jika seluruh komponen halaman ditampilkan dengan baik pada web browser yang diujikan. Sistem dianggap gagal jika terdapat komponen halaman yang tidak tampil atau ditampilkan tidak semestinya pada web browser yang diujikan. Berdasarkan review berbagai jenis web browser diambil 5 jenis terbaik untuk diujikan pada sistem ini. Ke-5 web browser tersebut adalah *Google Chrome*, *Mozilla Firefox*, *Internet Explorer*, *Opera* dan *Safari*

(<http://internet-browser-review.toptenreviews.com>). Hasil pengujian untuk ke-5 web browser tersebut adalah sebagai berikut.

Tabel 11 : *Test Case Google Chrome*

Test case id	Browser compatibility
Purpose	Menguji faktor <i>compatibility</i> dengan web browser Google Chrome
Assumptions	1. Sistem diakses pada web browser Google Chrome
Test data	1. login dengan browser google chrome 2. menampilkan data dengan google chrome
Steps	1. login sebagai admin /operator 2. klik menu entri → kode pelajaran
Expected result:	1. tampil halaman utama admin 2. tampil halaman data kode pelajaran
Actual result:	1. tampil halaman utama admin 2. tampil halaman data kode pelajaran
Pass/Fail:	Lolos

Hasil pengujian pada jenis web browser google chrome sesuai test case di atas dapat dilihat pada gambar berikut ini.



Gambar 16 : *Tampilan pada Google Chrome*

Pada gambar tersebut terlihat bahwa perangkat lunak dapat ditampilkan dengan baik pada web browser google chrome. Hasil pengujian pada jenis web browser

lain dapat dilihat pada halaman lampiran penelitian ini. Dari masing-masing jenis *web browser* yang diujikan, maka faktor *compatibility web browser* perangkat lunak ini dapat dikatakan LOLOS.

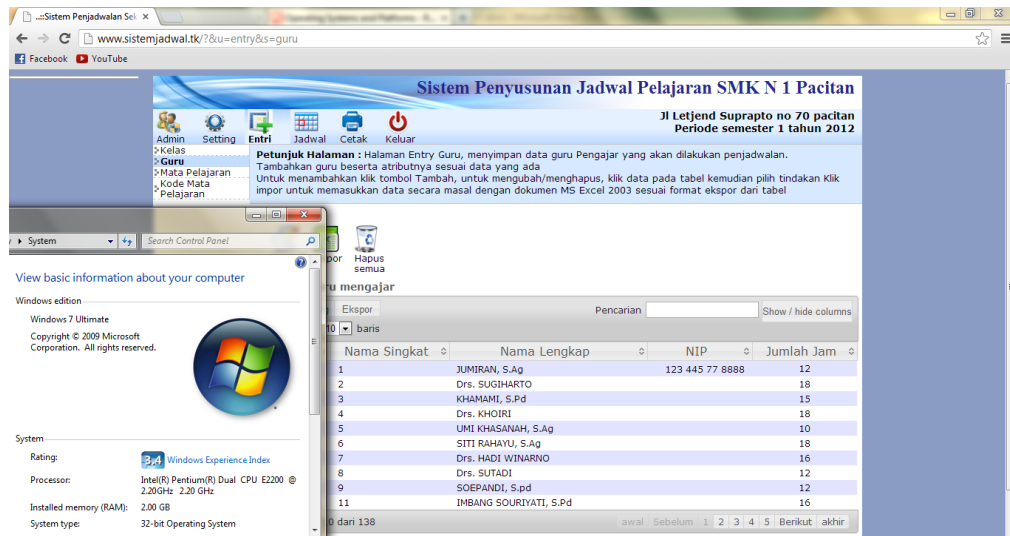
2) *Operating System Compatibility*

Pengujian faktor *compatibility* untuk jenis *Operating Sistem* dilakukan dengan mencoba menjalankan sistem pada beberapa sistem operasi yang sering digunakan. Pengujian dianggap lolos jika sistem dapat dijalankan dengan baik pada sistem operasi yang diujikan. Sistem dianggap gagal jika sistem tidak dapat dijalankan pada salah satu sistem operasi yang diujikan. Berikut hasil pengujian pada beberapa sistem operasi.

Tabel 12 : Hasil Pengujian *Operating System Compatibility*

Sistem operasi	Resolusi monitor yang digunakan					Keterangan
	800 x 600	1024 x 600	1024 x 768	1280 x 800	1366 x 768	
Ms Windows XP	V	V	V	V	V	Lolos
Ms Windows 7	V	V	V	V	V	Lolos
Ubuntu Lucid	V	V	V	V	V	Lolos

Alasan peneliti memilih ke-3 sistem operasi tersebut karena merupakan sistem operasi yang cenderung digunakan oleh pengguna perangkat lunak ini. Salah satu tampilan dari sistem operasi yang diujikan dapat dilihat pada gambar berikut.



Gambar 17 : Tampilan pada Windows 7

Pada gambar di atas terlihat perangkat lunak dijalankan pada sistem operasi Windows 7 dan berjalan baik. Hasil pengujian pada sistem operasi lainnya dapat juga dilihat pada halaman lampiran penelitian ini. Dari ke-3 jenis sistem operasi yang diujikan, maka dapat dikatakan faktor *compatibility operating system* perangkat lunak ini LOLOS.

3) *Printing Options*

Pengujian faktor *compatibility* untuk jenis *printer* dilakukan dengan mencoba mencetak jadwal pelajaran dengan beberapa jenis *printer* yang sering digunakan. Pengujian dianggap lolos jika seluruh halaman dapat dicetak dengan baik pada printer yang diujikan. Sistem dianggap gagal jika terdapat kesalahan pencetakan jadwal pada salah satu printer yang diujikan. Berikut hasil pengujian pada beberapa jenis printer.

Tabel 13 : Hasil Pengujian *Printing Compatibility*

Printer	Halaman yang dicetak			Keterangan
	Jadwal kelas	Jadwal guru	Jadwal utama	
Canon ip 1600	V	V	V	Lolos
Canon ip 2770	V	V	V	Lolos
Hp deskjet 3744	V	V	V	Lolos
Epson C45	V	V	V	Lolos
Hp laser 1020p	V	V	V	Lolos

Contoh hasil cetak untuk setiap jenis jadwal terlampir pada penelitian ini.

c. Pengujian faktor *user interface*.

Faktor *user interface* diuji dengan meneliti setiap komponen *user interface* kemudian membandingkan dengan standar yang ada. Berikut salah satu hasil pengujian untuk faktor *user interface*.

Tabel 14 : Test Case Screen Font Type

Test case id	<i>Screen Font Type</i>
Purpose	Menguji faktor <i>user interface</i> pada komponen jenis huruf (<i>font type</i>)
Assumptions	<ol style="list-style-type: none"> 1. Sistem diakses pada web browser 2. Login sebagai admin/operator
Test data	<ol style="list-style-type: none"> 1. Jenis huruf pada judul dan alamat sekolah 2. Jenis huruf pada menu dan submenu 3. Jenis huruf pada form dan tabel 4. Jenis huruf pada petunjuk dan pesan operasi
Steps	<ol style="list-style-type: none"> 1. Klik menu Entri → kode pelajaran 2. Klik menu Entri → guru
Expected result:	<ol style="list-style-type: none"> 1. Jenis huruf setiap komponen pada masing-masing halaman konsisten
Actual result:	<ol style="list-style-type: none"> 1. Jenis huruf judul dan alamat sekolah pada masing-masing halaman konsisten 2. Jenis huruf menu dan submenu pada masing-masing halaman konsisten 3. Jenis huruf form dan tabel pada masing-masing halaman konsisten 4. Jenis huruf petunjuk dan pesan operasi pada masing-masing halaman konsisten
Pass/Fail:	Lolos

Hasil pengujian faktor user interface selengkapnya dapat dilihat pada halaman lampiran penelitian ini. Berdasarkan hasil pengujian tersebut, setiap komponen yang diuji dan dibandingkan dengan standar aplikasi berbasis web dan semuanya sudah sesuai, maka faktor *user interface* untuk perangkat lunak ini dapat dikatakan LOLOS.

d. Pengujian faktor *usability*.

Pengujian faktor usability dilakukan dengan menyebarkan kuesioner. Berikut hasil pengujian yang telah dilakukan kepada 30 responden.

Tabel 15 : Rekapitulasi data kuesioner pengujian faktor *usability*

pertanyaan	Jawaban			
	Sangat setuju	Setuju	Kurang setuju	Tidak setuju
1	26	4	0	0
2	24	6	0	0
3	11	19	0	0
4	22	8	0	0
5	16	14	0	0
6	26	4	0	0
7	23	7	0	0
8	25	5	0	0
9	25	5	0	0
10	22	8	0	0
11	23	7	0	0
12	11	19	0	0
13	20	10	0	0
14	20	10	0	0
15	12	18	0	0
16	21	9	0	0
17	21	9	0	0
18	22	8	0	0
19	23	7	0	0
jumlah	393	177	0	0

B. Analisis Kualitas Perangkat Lunak.

Penyebaran kuesioner dalam penelitian ini dilakukan pada bulan Juni 2012 yang disebar sebanyak 30 lembar kuesioner yang terdiri dari 48 butir pertanyaan dan kembali sebanyak 30 lembar kuesioner (kembali semua).

1. Analisis faktor *User Interface*

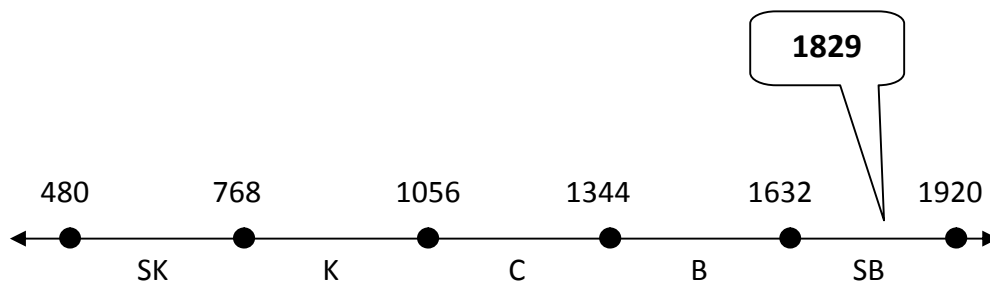
Berdasarkan hasil pengujian faktor *user interface*, semua *test case* termasuk pada kategori LOLOS. Didukung hasil perolehan kuesioner faktor *user interface* terdiri dari 16 butir pertanyaan dengan jumlah responden sebanyak 30 orang.

Tabel 16 : Rekapitulasi data kuesioner pengujian faktor *user interface*

pertanyaan	Jawaban			
	Sangat setuju	Setuju	Kurang setuju	Tidak setuju
1	27	3	0	0
2	28	2	0	0
3	27	3	0	0
4	28	2	0	0
5	25	5	0	0
6	29	1	0	0
7	22	8	0	0
8	25	5	0	0
9	20	10	0	0
10	23	7	0	0
11	24	6	0	0
12	26	4	0	0
13	24	6	0	0
14	25	5	0	0
15	16	14	0	0
16	20	10	0	0
jumlah	398	91	0	0

- Jumlah skor “Sangat Setuju” 398 x 4 = 1556
- Jumlah skor “Setuju” 91 x 3 = 273
- Jumlah Skor “Kurang Setuju” 0 x 2 = 0
- Jumlah Skor “Tidak Setuju” 0 x 1 = 0
- **Jumlah Total = 1829**

Maka hasilnya dapat dilihat sebagai berikut.



Gambar 18 : Perbandingan Nilai Hasil Kuesioner dengan Ketegorisasi Penilaian Faktor Kualitas *User Interface*

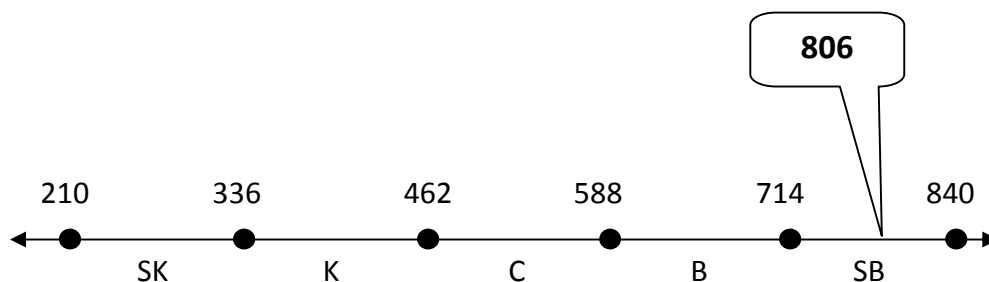
Berdasarkan hasil pengujian faktor *functionality* dengan *test case*, 28 test case dinyatakan LOLOS sehingga dapat dihitung sesuai rumus persentase $28:28 \times 100\% = 100\%$ maka sistem penyusunan jadwal pelajaran berbasis web di SMKN 1 Pacitan ini termasuk pada kategori LOLOS. Didukung dengan perolehan nilai dari kuesioner untuk faktor *functionality* terdiri dari 7 butir pertanyaan dari 30 responden dengan nilai total 806.

Tabel 17 : Rekapitulasi data kuesioner pengujian faktor *functionality*

pertanyaan	Jawaban			
	Sangat setuju	Setuju	Kurang setuju	Tidak setuju
1	27	3	0	0
2	24	6	0	0
3	25	5	0	0
4	25	5	0	0
5	24	6	0	0
6	27	3	0	0
7	24	6	0	0
jumlah	176	34	0	0

- Jumlah skor “Sangat Setuju” $176 \times 4 = 704$
- Jumlah skor “Setuju” $34 \times 3 = 102$
- Jumlah Skor “Kurang Setuju” $0 \times 2 = 0$
- Jumlah Skor “Tidak Setuju” $0 \times 1 = 0$
- **Jumlah Total = 806**

Perolehan nilai total tersebut selanjutnya dibandingkan dengan skala pengukuran 5 kelas. Maka hasilnya dapat dilihat sebagai berikut.



Gambar 19 : Perbandingan Nilai Hasil Kuesioner dengan Kategorisasi Penilaian Faktor Kualitas *functionality*

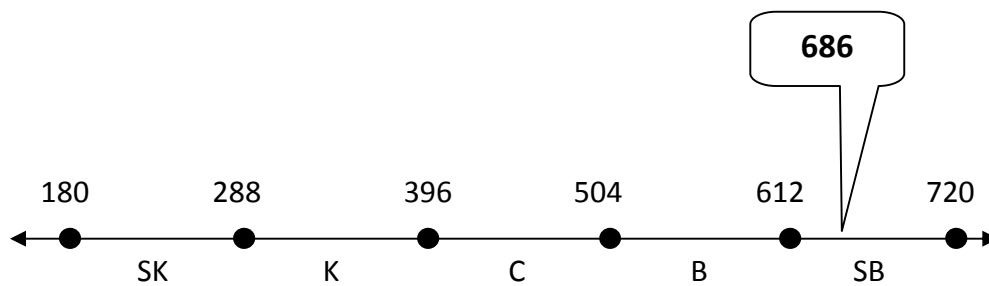
3. Analisis faktor *compatibility*

Berdasarkan hasil pengujian faktor compatibility terdiri dari *browser compatibility*, *operating system compatibility* dan *printing compatibility* dapat dikategorikan LOLOS. Dari 5 jenis *web browser* yang diujikan masing-masing dapat berjalan dengan baik. Kemudian dari 3 sistem operasi yang diujikan masing-masing juga berjalan dengan baik. Yang terakhir pengujian printing, dari 5 jenis printer yang diujikan untuk 3 halaman cetak masing-masing juga berjalan baik. Analisis ini didukung dengan perolehan nilai kuesioner dari 6 butir pertanyaan dengan jumlah responden sebanyak 30 orang diperoleh skor total 686. Sehingga skor tersebut termasuk dalam kategori sangat baik.

Tabel 40 : Rekapitulasi data kuesioner pengujian faktor *compatibility*

pertanyaan	Jawaban			
	Sangat setuju	Setuju	Kurang setuju	Tidak setuju
1	23	7	0	0
2	22	8	0	0
3	23	7	0	0
4	26	4	0	0
5	24	6	0	0
6	28	2	0	0
jumlah	146	34	0	0

- | | | |
|-------------------------------------|--------------|--------------|
| • Jumlah skor “Sangat Setuju” | 146 x 4 | = 584 |
| • Jumlah skor “Setuju” | 34 x 3 | = 102 |
| • Jumlah Skor “Kurang Setuju” | 0 x 2 | = 0 |
| • <u>Jumlah Skor “Tidak Setuju”</u> | <u>0 x 1</u> | <u>= 0</u> |
| • Jumlah Total | | = 686 |



Gambar 20 : Perbandingan Nilai Hasil Kuesioner dengan Kategorisasi Penilaian Faktor Kualitas *compatibility*

4. Analisis faktor *usability*

Analisis faktor *usability* didapatkan dari perolehan nilai kuesioner. Dari 19 item pertanyaan untuk faktor *usability* dengan 30 responden, diperoleh skor total 2013.

- Jumlah skor “Sangat Setuju” $393 \times 4 = 1572$
- Jumlah skor “Setuju” $177 \times 3 = 531$
- Jumlah Skor “Kurang Setuju” $0 \times 2 = 0$
- Jumlah Skor “Tidak Setuju” $0 \times 1 = 0$
- **Jumlah Total = 2013**

Skor tersebut termasuk pada kategori sangat baik yaitu pada range 1938 s.d 2280 dapat dilihat pada gambar berikut.



Gambar 21 : Perbandingan Nilai Hasil Kuesioner dengan Kategorisasi Penilaian Faktor Kualitas *usability*

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan pembahasan yang telah diuraikan berikutnya dapat ditarik kesimpulan sebagai berikut :

1. Hasil pengembangan perangkat lunak Sistem penyusunan jadwal pelajaran sekolah berbasis web dapat digunakan dengan baik di SMKN 1 Pacitan.
2. Seluruh fungsi yang diperlukan berjalan dengan baik sehingga dapat disimpulkan perangkat lunak ini memenuhi standar kualitas *functionality*. Didukung dengan hasil kuesioner yang menyatakan bahwa kualitas faktor *functionality* perangkat lunak ini pada kategori sangat baik.
3. Perangkat lunak ini dapat dijalankan pada berbagai jenis web browser, sistem operasi, dan berbagai jenis printer untuk mencetak. Didukung dengan hasil kuesioner untuk faktor *compatibility* pada kategori sangat baik. Sehingga dapat ditarik kesimpulan bahwa perangkat lunak ini memenuhi standar kualitas *compatibility*.
4. Setiap komponen *user interface* sudah sesuai dengan standar pengembangan user interface aplikasi berbasis web. Didukung dengan hasil kuesioner untuk faktor user interface pada kategori sangat baik. Sehingga dapat disimpulkan perangkat lunak ini memenuhi standar kualitas *user interface* aplikasi berbasis web.

5. Hasil analisis faktor *usability* berdasarkan kuesioner menunjukkan bahwa perangkat lunak ini pada kategori sangat baik

B. Keterbatasan Penelitian

Walaupun penelitian ini telah diusahakan sebaik-baiknya, tetapi tidak lepas dari keterbatasan dan kekurangan antara lain :

1. Penelitian ini hanya berlaku kepada sekolah SMKN 1 Pacitan.
2. Penelitian ini hanya meneliti empat faktor kualitas perangkat lunak. Masih ada faktor lain yang belum terjangkau dalam kuesioner penelitian ini.

C. Saran

Hendaknya selain penelitian ini, perlu adanya penelitian-penelitian lain mengenai kualitas perangkat lunak berbasis web dengan tujuan yang sama yaitu untuk kepentingan dan pengembangan perangkat lunak Sistem Penyusunan Jadwal Pelajaran Berbasis Web. Peneliti sangat mengharapkan kepada pihak yang terkait baik sekolah atau lembaga pemerintahan agar dapat memberikan perhatian dan dukungan lebih untuk mengembangkan perangkat lunak tersebut agar lebih sempurna.

D. Implikasi

Hasil dari penelitian ini dapat dijadikan sebagai masukan sehingga dapat diambil langkah yang tepat untuk mengembangkan perangkat lunak Sistem Penyusunan Jadwal pelajaran Berbasis Web.

DAFTAR PUSTAKA

- Agarwal, B. B., Tayal, S. P., & Gupta, M. (2010). *Software Engineering and Testing*. Sudbury: Jones and Bartlett Publishers.
- Alain Abran. Et al.(2004). *Guide to the Software Engineering Body of Knowledge*. USA : Angela Burgess
- Achmad Basuki. (2003). ALGORITMA GENETIKA : Suatu Alternatif Penyelesaian Permasalahan Searching, Optimasi dan Machine Learning. *Diktat*, Surabaya : PENS ITS
- Cecep Susan Jumena. (2006). Penyelesaian Masalah Penjadwalan Kuliah Menggunakan Algoritma *Tabusearch*. *Skripsi*. Teknik Informatika UKI.
- Champlain, J. J. (2003). *Auditing information systems*. Canada: John Wiley & Sons, Inc.
- Eko Zulkaryanto dan Mushthofa.(2011). Sistem Penjadwalan Ujian Menggunakan Answer Set Programming. *Jurnal MIPA IPB*. Hlm 1-10
- Gustafson, David A. (2002). *Theory and Problems of SOFTWARE ENGINEERING*. New York : McGraw-Hill
- Hanif Al Fatta. (2007). *Analisis & Perancangan Sistem Informasi*. Yogyakarta : Andi Offset
- Laplante, P. A. (2004). *Real-time systems design and analysis : an engineer's handbook*. Canada: A JOHN WILEY & SONS, INC.
- O'Brien, J. A., & Marakas, G. M. (2010). *Introduction to information systems*. New York: McGraw-Hill.
- Petrovic, Burke. (2006). A Graph-Based Hyper-Heuristic for Educational Timetabling Problems. *Paper*, European Journal of Operational Research. Hlm 2
- Pressman, R. (2001). *Software Engineering*. New York : McGraw-Hill
- Sheldon, R. (2005). *Beginning MySQL*. New York : Wiley Publishing
- Simarmata Janner. (2010). *Rekayasa Perangkat Lunak*. Yogyakarta : Andi Offset
- Sugiyono. (2010). *Metode Penelitian Bisnis*. Bandung : Alfabeta

- Sugiyono. (2007). *Statistik Untuk Penelitian*. Bandung : Alfabeta
- Sugiyono. (2012). *Metode Penelitian Kuantitatif Kualitatif Dan R&D*. Bandung : Alfabeta
- Vaswani, V. (2007). *PHP Programming Solutions*. New York : McGraw-Hill
- Yana Maulana. (2006) . *Penjadwalan Perkuliahan Dengan Menggunakan Algoritma Max-Min Ant System*. Jurnal Teknik Informatika UNIKOM. Hlm 1-6.
- <http://www.php.net>
- <http://blog.smartbear.com/post/09-12-22/15-useful-test-cases-for-ensuring-consistent-user-interfaces/>
- <http://www.testinggeek.com/web-application-interface-and-compatibility-checklist>
- <http://hcibib.org/perlman/question.cgi>
- <http://www.softwaretestinghelp.com/web-application-testing>
- <http://bahasa.kemdiknas.go.id>
- <http://internet-browser-review.toptenreviews.com>
- <http://www.asctimetables.com>
- <http://www.asctt.com/ttexport/>

LAMPIRAN

UJI VALIDITAS DAN RELIABILITAS KUESIONER

Item-Total Statistics

	Scale Mean if Item Deleted	Scale Variance if Item Deleted	Corrected Item-Total Correlation	Cronbach's Alpha if Item Deleted
x1	185,5000	126,500	,828	,953
x2	185,5000	126,500	,828	,953
x3	185,5000	126,500	,828	,953
x4	185,5000	130,278	,292	,955
x5	185,6000	129,378	,306	,955
x6	185,5000	130,278	,292	,955
x7	185,7000	128,456	,347	,955
x8	185,6000	124,267	,856	,952
x9	185,8000	123,956	,719	,953
x10	185,7000	126,233	,555	,954
x11	185,6000	127,156	,542	,954
x12	185,5000	126,500	,828	,953
x13	185,6000	127,156	,542	,954
x14	185,6000	127,156	,542	,954
x15	185,9000	128,544	,307	,955
x16	185,8000	127,956	,365	,955
x17	185,5000	130,278	,292	,955
x18	185,6000	129,378	,306	,955
x19	185,6000	129,378	,306	,955
x20	185,6000	124,267	,856	,952
x21	185,6000	124,267	,856	,952
x22	185,5000	126,500	,828	,953
x23	185,7000	132,011	,022	,956
x24	185,6000	124,267	,856	,952
x25	185,6000	129,378	,306	,955
x26	185,7000	126,011	,576	,953
x27	185,7000	127,789	,409	,954
x28	185,6000	124,267	,856	,952
x29	185,7000	124,900	,681	,953
x30	185,5000	126,500	,828	,953
x31	185,5000	126,500	,828	,953
x32	185,6000	127,600	,495	,954
x33	185,9000	125,211	,593	,953
x34	185,6000	125,600	,710	,953
x35	185,8000	127,511	,404	,954

x36	185,5000	126,500	,828	,953
x37	185,6000	128,489	,400	,954
x38	185,5000	130,278	,292	,955
x39	185,6000	130,711	,166	,955
x40	185,5000	126,500	,828	,953
x41	185,6000	129,378	,306	,955
x42	185,5000	130,278	,292	,955
x43	185,9000	127,656	,383	,955
x44	185,5000	126,500	,828	,953
x45	185,5000	126,500	,828	,953
x46	185,8000	127,067	,443	,954
x47	185,5000	126,500	,828	,953
x48	185,6000	127,600	,495	,954
x49	185,6000	128,489	,400	,954
x50	185,6000	127,156	,542	,954

Reliability Statistics

Cronbach's	
Alpha	N of Items
,955	50

HASIL PENGUJIAN FAKTOR FUNCTIONALITY

Tabel 1 : *Test Case Insert*

Test case id	Insert (CRUD)
Purpose	Menguji fungsi tambah data (INSERT)
Assumptions	<ol style="list-style-type: none"> 1. Sistem diakses pada web browser 2. login sebagai admin / operator
Test data	<ol style="list-style-type: none"> 1. Berhasil menyimpan 2. Gagal menyimpan
Steps	<ol style="list-style-type: none"> 1. Klik menu entri → kelas → tambah → isi semua field → simpan 2. Klik menu entri → kelas → tambah → kosongkan salah satu field → simpan
Expected result:	<ol style="list-style-type: none"> 1. Muncul pesan “operasi berhasil”, data baru masuk pada tabel 2. Muncul pesan “Kesalahan : Semua input harus diisi”
Actual result:	<ol style="list-style-type: none"> 1. Muncul pesan “operasi berhasil”, data baru masuk pada tabel 2. Muncul pesan “Kesalahan : Semua input harus diisi”
Pass/Fail:	Lolos

Tabel 2 : *Test Case Update*

Test case id	Update (CRUD)
Purpose	Menguji fungsi edit data (UPDATE)
Assumptions	<ol style="list-style-type: none"> 1. Sistem diakses pada web browser 2. login sebagai admin / operator
Test data	<ol style="list-style-type: none"> 1. Berhasil menyimpan 2. Gagal menyimpan
Steps	<ol style="list-style-type: none"> 1. Klik menu entri → kelas → klik salah satu data pada tabel → pilih menu edit → ubah beberapa field → simpan 2. Klik menu entri → kelas → klik salah satu data pada tabel → pilih menu edit → kosongkan beberapa field → simpan
Expected result:	<ol style="list-style-type: none"> 1. Muncul pesan “operasi berhasil”, data pada tabel berubah 2. Muncul pesan “Kesalahan : Semua input harus diisi”
Actual result:	<ol style="list-style-type: none"> 1. Muncul pesan “operasi berhasil”, data pada tabel berubah 2. Muncul pesan “Kesalahan : Semua input harus diisi”
Pass/Fail:	Lolos

Tabel 3 : *Test Case Delete*

Test case id	Delete (CRUD)
Purpose	Menguji fungsi hapus data (DELETE)
Assumptions	<ol style="list-style-type: none"> 1. Sistem diakses pada web browser 2. login sebagai admin /operator 3. salah satu kelas sudah dijadwalkan
Test data	<ol style="list-style-type: none"> 1. Berhasil menghapus 2. Gagal menghapus
Steps	<ol style="list-style-type: none"> 1. Klik menu entri → kelas → klik kelas yang belum dijadwalkan pada tabel →pilih menu delete→ klik OK 2. Klik menu entri → kelas → klik kelas yang sudah dijadwalkan pada tabel →pilih menu delete→ klik OK
Expected result:	<ol style="list-style-type: none"> 1. Muncul pesan “operasi berhasil”, data pada tabel hilang 2. Muncul pesan “Operasi gagal”, data pada tabel tetap ada
Actual result:	<ol style="list-style-type: none"> 1. Muncul pesan “operasi berhasil”, data pada tabel hilang 2. Muncul pesan “Operasi gagal”, data pada tabel tetap ada
Pass/Fail:	Lolos

Tabel 4 : *Test Case Export*

Test case id	Export
Purpose	Menguji fungsi ekspor data
Assumptions	<ol style="list-style-type: none"> 1. Sistem diakses pada web browser 2. login sebagai admin /operator 3. data kelas sudah dimasukkan
Test data	<ol style="list-style-type: none"> 1. Berhasil meng-ekspor data
Steps	<ol style="list-style-type: none"> 1. Klik menu entri → kelas → klik tombol ekspor pada tabel →pilih Excel→tuliskan nama file→klik Save
Expected result:	<ol style="list-style-type: none"> 1. Muncul file excel pada folder dan berisi data kelas
Actual result:	<ol style="list-style-type: none"> 1. Muncul file excel pada folder dan berisi data kelas
Pass/Fail:	Lolos

Tabel 5 : *Test Case Import*

Test case id	Import
Purpose	Menguji fungsi impor data
Assumptions	<ol style="list-style-type: none"> 1. Sistem diakses pada web browser 2. login sebagai admin / operator 3. data kelas sudah dientri pada file Excel 2003 (xls) dengan format sesuai ekspor data 4. data kelas sudah dientri pada file Excel 2003 (xls) dengan format yang berbeda dengan ekspor data
Test data	<ol style="list-style-type: none"> 1. Berhasil meng-impor data 2. Gagal meng-impor data
Steps	<ol style="list-style-type: none"> 1. Klik menu entri → kelas → klik tombol impor → pilih file Excel dengan format yang benar → klik Simpan 2. Klik menu entri → kelas → klik tombol impor → pilih file Excel dengan format yang salah → klik Simpan
Expected result:	<ol style="list-style-type: none"> 1. Muncul pesan “operasi berhasil”, data kelas bertambah 2. Muncul pesan “Isi File Salah, pastikan format sesuai dengan file ekspor dari data kelas”
Actual result:	<ol style="list-style-type: none"> 1. Muncul pesan “operasi berhasil”, data kelas bertambah 2. Muncul pesan “Isi File Salah, pastikan format sesuai dengan file ekspor dari data kelas”
Pass/Fail:	Lolos

Tabel 6 : *Test Case Search*

Test case id	Search
Purpose	Menguji fungsi pencarian data
Assumptions	<ol style="list-style-type: none"> 1. Sistem diakses pada web browser 2. login sebagai admin /operator 3. data kelas sudah dimasukkan
Test data	<ol style="list-style-type: none"> 1. Berhasil mencari data 2. Gagal mencari data
Steps	<ol style="list-style-type: none"> 1. Klik menu entri → kelas → masukkan nama kelas yang ada 2. Klik menu entri → kelas → masukkan nama kelas yang tidak ada
Expected result:	<ol style="list-style-type: none"> 1. Muncul data kelas yang dicari pada tabel 2. Muncul pesan “data kosong” pada tabel
Actual result:	<ol style="list-style-type: none"> 1. Muncul data kelas yang dicari pada tabel 2. Muncul pesan “data kosong” pada tabel
Pass/Fail:	Lolos

Tabel 7 : *Test Case Auto generate*

Test case id	Auto generate
Purpose	Menguji fungsi auto generata jadwal
Assumptions	<ol style="list-style-type: none"> 1. Sistem diakses pada web browser 2. login sebagai admin /operator 3. data kelas , mapel, guru, kode pelajaran sudah dimasukkan 4. sudah ada kelas yang jam 1-4 terisi pertemuan, misalnya kelas X MM 2
Test data	<ol style="list-style-type: none"> 1. Berhasil mengalokasikan jadwal 2. Gagal mengalokasikan jadwal
Steps	<ol style="list-style-type: none"> 1. Klik menu jadwal→ jadwal kelas → tambahkan pertemuan untuk kelas yang belum penuh 2. Klik menu jadwal→ jadwal kelas → tambahkan pertemuan untuk kelas X MM 2 dengan batas jam ke 4 3. Klik menu auto generate
Expected result:	<ol style="list-style-type: none"> 1. Muncul pesan “5 item berhasil dialokasikan ke jadwal” 2. Muncul pesan “4 item gagal dialokasikan ke jadwal”
Actual result:	<ol style="list-style-type: none"> 1. Muncul pesan “5 item berhasil dialokasikan ke jadwal” 2. Muncul pesan “4 item gagal dialokasikan ke jadwal”
Pass/Fail:	Lolos

Tabel 8 : *Test Case Manual Schedule*

Test case id	Manual schedule
Purpose	Menguji fungsi tambah pertemuan jadwal secara manual
Assumptions	<ol style="list-style-type: none"> 1. Sistem diakses pada web browser 2. login sebagai admin /operator 3. data kelas , mapel, guru, kode pelajaran sudah dimasukkan 4. sudah ada guru yang mengajar pada jam ke 2 hari senin, misalnya JUMIRAN
Test data	<ol style="list-style-type: none"> 1. Berhasil menambahkan jadwal 2. Gagal menambahkan jadwal
Steps	<ol style="list-style-type: none"> 1. Klik menu jadwal utama→klik pada slot kosong matriks jadwal→klik tambah → pilih kode pelajaran →klik simpan. 2. Klik menu jadwal utama→klik pada slot kosong matriks jadwal pada hari senin jam ke 2→klik tambah → pilih kode pelajaran untuk guru JUMIRAN→klik simpan
Expected result:	<ol style="list-style-type: none"> 1. Muncul pesan “operasi berhasil”, pertemuan pada jadwal bertambah 2. Muncul pesan “jadwal bentrok dengan kelas lain”
Actual result:	<ol style="list-style-type: none"> 1. Muncul pesan “operasi berhasil”, pertemuan pada jadwal bertambah 2. Muncul pesan “jadwal bentrok dengan kelas lain”
Pass/Fail:	Lolos

Tabel 9 : *Test Case Print*

Test case id	Print
Purpose	Menguji fungsi cetak/print jadwal
Assumptions	<ol style="list-style-type: none"> 1. Sistem diakses pada web browser 2. login sebagai admin /operator 3. data kelas , mapel, guru, kode pelajaran sudah dimasukkan 4. sudah ada minimal 1 kelas yang dijadwalkan, misalnya kelas XMM2 5. sudah ada minimal 1 guru yang dijadwalkan, misalnya kelas JUMIRAN
Test data	<ol style="list-style-type: none"> 1. Berhasil mencetak jadwal (kelas) 2. Berhasil mencetak jadwal (guru) 3. Berhasil mencetak jadwal (utama)
Steps	<ol style="list-style-type: none"> 1. Klik menu cetak→klik cetak jadwal per kelas→pilih nama kelas XMM2→ klik proses→klik cetak sekarang. 2. Klik menu cetak→klik cetak jadwal per guru→pilih nama guru JUMIRAN→ klik proses→klik cetak sekarang 3. Klik menu cetak→klik cetak semua jadwal →pilih tingkat kelas dan warna→ klik proses→klik cetak sekarang
Expected result:	<ol style="list-style-type: none"> 1. Muncul file PDF dengan isi jadwal pelajaran untuk kelas XMM2 2. Muncul file PDF dengan isi jadwal mengajar untuk guru JUMIRAN 3. Muncul file PDF dengan isi jadwal pelajaran dari senin sampai sabtu untuk semua kelas
Actual result:	<ol style="list-style-type: none"> 1. Muncul file PDF dengan isi jadwal pelajaran untuk kelas XMM2 2. Muncul file PDF dengan isi jadwal mengajar untuk guru JUMIRAN 3. Muncul file PDF dengan isi jadwal pelajaran dari senin sampai sabtu untuk semua kelas
Pass/Fail:	Lolos

HASIL PENGUJIAN FAKTOR USER INTERFACE

Tabel 10 : Test Case Screen Font Size

Test case id	<i>Screen Font Sizes</i>
Purpose	Menguji faktor <i>user interface</i> pada komponen ukuran huruf (<i>font size</i>)
Assumptions	<ol style="list-style-type: none"> 1. Sistem diakses pada web browser 2. Login sebagai admin/operator
Test data	<ol style="list-style-type: none"> 1. Ukuran huruf pada judul dan alamat sekolah 2. Ukuran huruf pada menu dan submenu 3. Ukuran huruf pada form dan tabel 4. Ukuran huruf pada petunjuk dan pesan operasi
Steps	<ol style="list-style-type: none"> 1. Klik menu Entri → kode pelajaran 2. Klik menu Entri → guru
Expected result:	<ol style="list-style-type: none"> 1. Ukuran huruf setiap komponen pada masing-masing halaman konsisten
Actual result:	<ol style="list-style-type: none"> 1. Ukuran huruf judul dan alamat sekolah pada masing-masing halaman konsisten 2. Ukuran huruf menu dan submenu pada masing-masing halaman konsisten 3. Ukuran huruf form dan tabel pada masing-masing halaman konsisten 4. Jenis huruf petunjuk dan pesan operasi pada masing-masing halaman konsisten
Pass/Fail:	Lolos

Tabel 11 : Test Case Colors

Test case id	<i>Colors</i>
Purpose	Menguji faktor <i>user interface</i> pada komponen warna (<i>Colors</i>)
Assumptions	<ol style="list-style-type: none"> 1. Sistem diakses pada web browser 2. Login sebagai admin/operator
Test data	<ol style="list-style-type: none"> 1. Warna huruf pada judul, menu, submenu, form dan tabel data 2. Warna <i>background</i> pada judul, menu, submenu, form dan tabel data
Steps	<ol style="list-style-type: none"> 1. Klik menu Entri → kode pelajaran 2. Klik menu Entri → guru
Expected result:	<ol style="list-style-type: none"> 1. Warna setiap komponen pada masing-masing halaman konsisten
Actual result:	<ol style="list-style-type: none"> 1. Warna huruf pada judul, menu, submenu, form dan tabel data setiap halaman konsisten 2. Warna <i>background</i> pada judul, menu, submenu, form dan tabel data setiap halaman konsisten
Pass/Fail:	Lolos

Tabel 12 : Test Case Icons

Test case id	<i>Icons</i>
Purpose	Menguji faktor <i>user interface</i> pada komponen ikon (<i>Icons</i>)
Assumptions	<ol style="list-style-type: none"> 1. Sistem diakses pada web browser 2. Login sebagai admin/operator
Test data	<ol style="list-style-type: none"> 1. Icon pada menu dan toolbar
Steps	<ol style="list-style-type: none"> 1. Klik menu Entri → kode pelajaran 2. Klik menu Entri → guru
Expected result:	<ol style="list-style-type: none"> 1. Icon setiap komponen pada masing-masing halaman konsisten
Actual result:	<ol style="list-style-type: none"> 1. Icon pada menu dan toolbar pada masing-masing halaman konsisten
Pass/Fail:	Lolos

Tabel 13 : Test Case Narrative Text

Test case id	<i>Narrative Text</i>
Purpose	Menguji faktor <i>user interface</i> pada komponen teks naratif(<i>Narrative Text</i>)
Assumptions	<ol style="list-style-type: none"> 1. Sistem diakses pada web browser 2. Login sebagai admin/operator
Test data	<ol style="list-style-type: none"> 1. Teks naratif pada petunjuk halaman 2. Teks naratif pada pesan operasi 3. Teks naratif pada peringatan penghapusan
Steps	<ol style="list-style-type: none"> 1. Klik menu Entri → kode pelajaran 2. Klik menu Entri → guru
Expected result:	<ol style="list-style-type: none"> 1. Setiap halaman terdapat teks naratif sebagai petunjuk penggunaan
Actual result:	<ol style="list-style-type: none"> 1. Setiap halaman selalu terdapat teks naratif sebagai petunjuk penggunaan
Pass/Fail:	Lolos

Tabel 14 : Test Case Brevity

Test case id	<i>Brevity</i>
Purpose	Menguji faktor <i>user interface</i> pada komponen keringkasan pesan (<i>Brevity</i>)
Assumptions	<ol style="list-style-type: none"> 1. Sistem diakses pada web browser 2. Login sebagai admin/operator 3. Sudah ada guru yang dijadwalkan
Test data	<ol style="list-style-type: none"> 1. Pesan kesalahan penambahan data 2. Pesan kesalahan penghapusan data
Steps	<ol style="list-style-type: none"> 1. Klik menu Entri → kode pelajaran → klik tambah → kosongkan salah satu field → simpan 2. Klik menu Entri → guru → pilih guru yang sudah dijadwalkan → klik delete → klik OK
Expected result:	<ol style="list-style-type: none"> 1. Penyimpanan gagal dan muncul pesan kesalahan 2. Penghapusan gagal dan muncul pesan kesalahan
Actual result:	<ol style="list-style-type: none"> 1. Muncul pesan “semua field harus diisi”. 2. Muncul pesan “operasi gagal”
Pass/Fail:	Lolos

Tabel 15 : Test Case Dialog Box Consistency

Test case id	<i>Dialog Box Consistency</i>
Purpose	Menguji faktor <i>user interface</i> pada komponen tombol <i>dialog box</i> (<i>Dialog Box Consistency</i>)
Assumptions	<ol style="list-style-type: none"> 1. Sistem diakses pada web browser 2. Login sebagai admin/operator
Test data	1. Tombol pilihan pada <i>dialog box</i>
Steps	<ol style="list-style-type: none"> 1. Klik menu Entri → guru → pilih salah satu guru → klik delete 2. Klik menu Entri → kelas → pilih salah satu kelas → klik delete
Expected result:	1. Urutan tombol jawaban konsisten
Actual result:	<ol style="list-style-type: none"> 1. Urutan tombol jawaban Oke lalu Batal pada halaman guru 2. Urutan tombol jawaban Oke lalu Batal pada halaman kelas
Pass/Fail:	Lolos

Tabel 16 : Test Case Links

Test case id	<i>Links</i>
Purpose	Menguji faktor <i>user interface</i> pada komponen tombol <i>Link</i>
Assumptions	<ol style="list-style-type: none"> 1. Sistem diakses pada web browser 2. Login sebagai admin/operator
Test data	<ol style="list-style-type: none"> 1. Posisi menu, submenu dan toolbar 2. Urutan menu, submenu dan toolbar 3. Warna menu, submenu dan toolbar
Steps	<ol style="list-style-type: none"> 1. Klik menu Entri → guru 2. Klik menu Entri → kelas
Expected result:	<ol style="list-style-type: none"> 1. Posisi menu, submenu dan toolbar konsisten 2. Urutan menu, submenu dan toolbar konsisten 3. Warna menu, submenu dan toolbar konsisten
Actual result:	<ol style="list-style-type: none"> 1. Posisi menu, submenu dan toolbar pada setiap halaman konsisten 2. Urutan menu, submenu dan toolbar pada setiap halaman konsisten 3. Warna menu, submenu dan toolbar pada setiap halaman konsisten
Pass/Fail:	Lolos

Tabel 17 : Test Case Delete Confirmations

Test case id	<i>Delete Confirmations</i>
Purpose	Menguji faktor <i>user interface</i> pada komponen konfirmasi penghapusan data (<i>Delete Confirmations</i>)
Assumptions	<ol style="list-style-type: none"> 1. Sistem diakses pada web browser 2. Login sebagai admin/operator
Test data	<ol style="list-style-type: none"> 1. Konfirmasi penghapusan data tunggal 2. Konfirmasi penghapusan data masal
Steps	<ol style="list-style-type: none"> 1. Klik menu Entri → guru → pilih salah satu guru → klik delete 2. Klik menu Entri → kelas → klik Hapus semua
Expected result:	1. Muncul konfirmasi penghapusan pada setiap operasi hapus
Actual result:	<ol style="list-style-type: none"> 1. Muncul dialog konfirmasi “yakin menghapus data ini?” 2. Muncul dialog konfirmasi “yakin mengosongkan data kelas?”
Pass/Fail:	Lolos

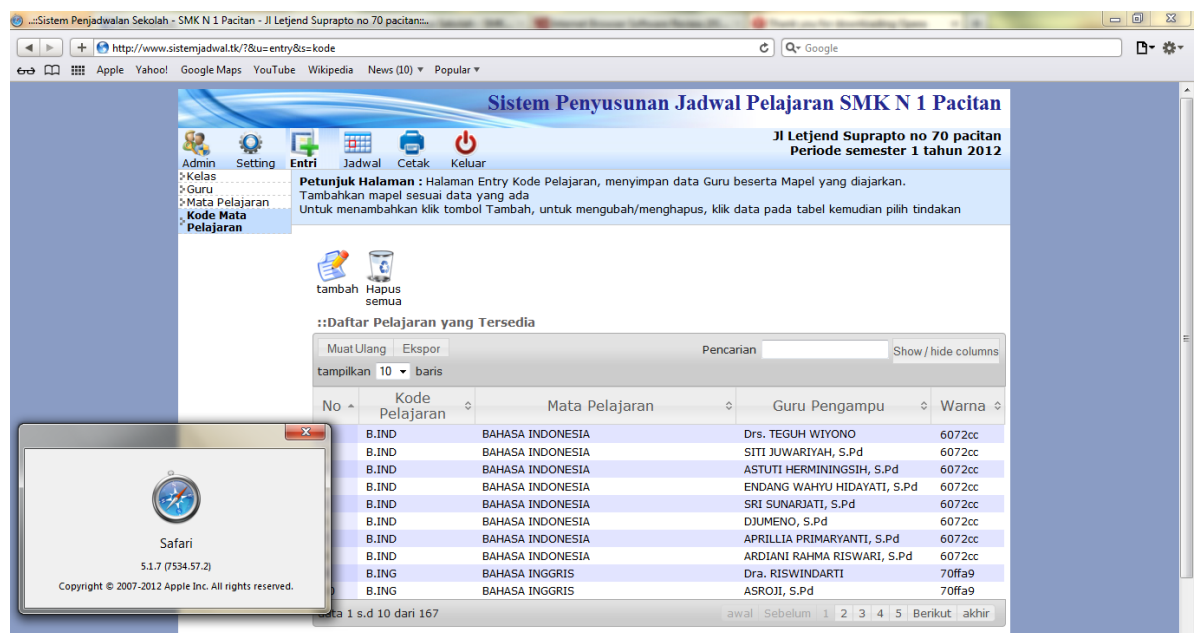
Tabel 18 : Test Case Save Confirmations

<i>Test case id</i>	<i>Save Confirmations</i>
<i>Purpose</i>	Menguji faktor <i>user interface</i> pada komponen konfirmasi penyimpanan data (<i>Save Confirmations</i>)
<i>Assumptions</i>	<ol style="list-style-type: none"> 1. Sistem diakses pada web browser 2. Login sebagai admin/operator
<i>Test data</i>	<ol style="list-style-type: none"> 1. Konfirmasi penyimpanan jadwal pelajaran
<i>Steps</i>	<ol style="list-style-type: none"> 1. Klik menu jadwal→ auto generate→ klik OK → klik simpan ke jadwal
<i>Expected result:</i>	<ol style="list-style-type: none"> 1. Muncul konfirmasi penyimpanan data
<i>Actual result:</i>	<ol style="list-style-type: none"> 1. Muncul dialog konfirmasi “apakah anda yakin menyimpan ke Jadwal utama?”
<i>Pass/Fail:</i>	Lolos

HASIL PENGUJIAN FAKTOR COMPATIBILITY (WEB BROWSER)

Tabel 19 : *Test Case Safari*

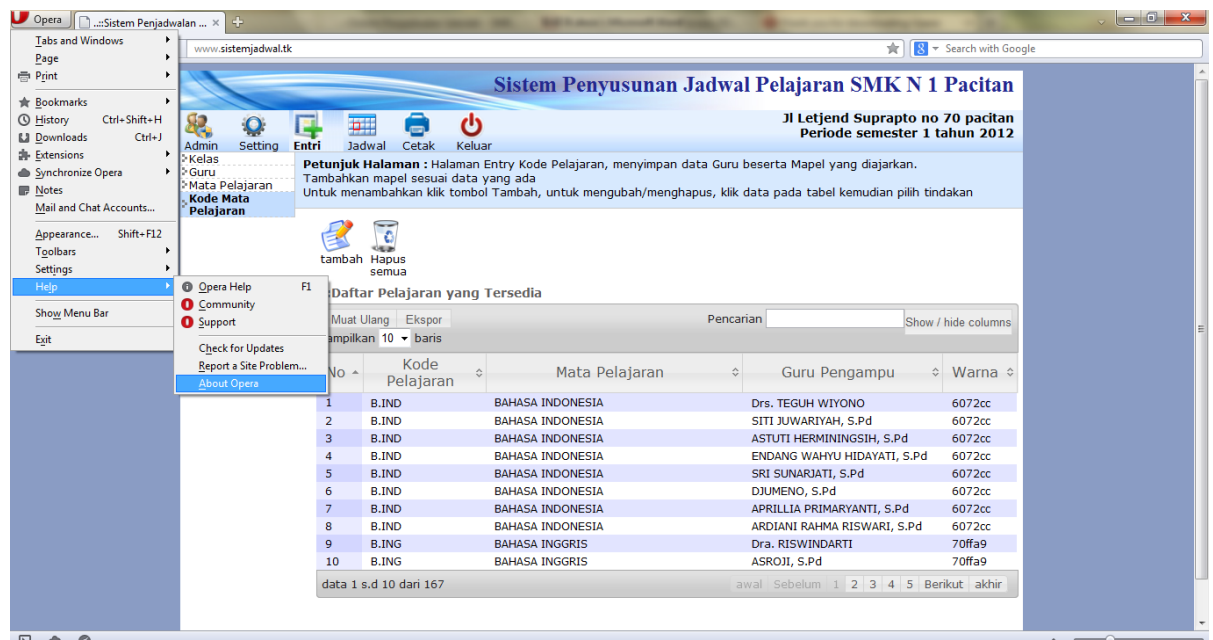
Test case id	Browser compatibility
Purpose	Menguji faktor <i>compatibility</i> dengan <i>web browser Safari</i>
Assumptions	1. Sistem diakses pada web browser Safari
Test data	1. login dengan browser Safari 2. menampilkan data dengan Safari
Steps	1. login sebagai admin /operator 2. klik menu entri → kode pelajaran
Expected result:	1. tampil halaman utama admin 2. tampil halaman data kode pelajaran
Actual result:	1. tampil halaman utama admin 2. tampil halaman data kode pelajaran
Pass/Fail:	Lolos



Gambar 1 : Tampilan pada *Safari*

Tabel 20 : *Test Case Opera*

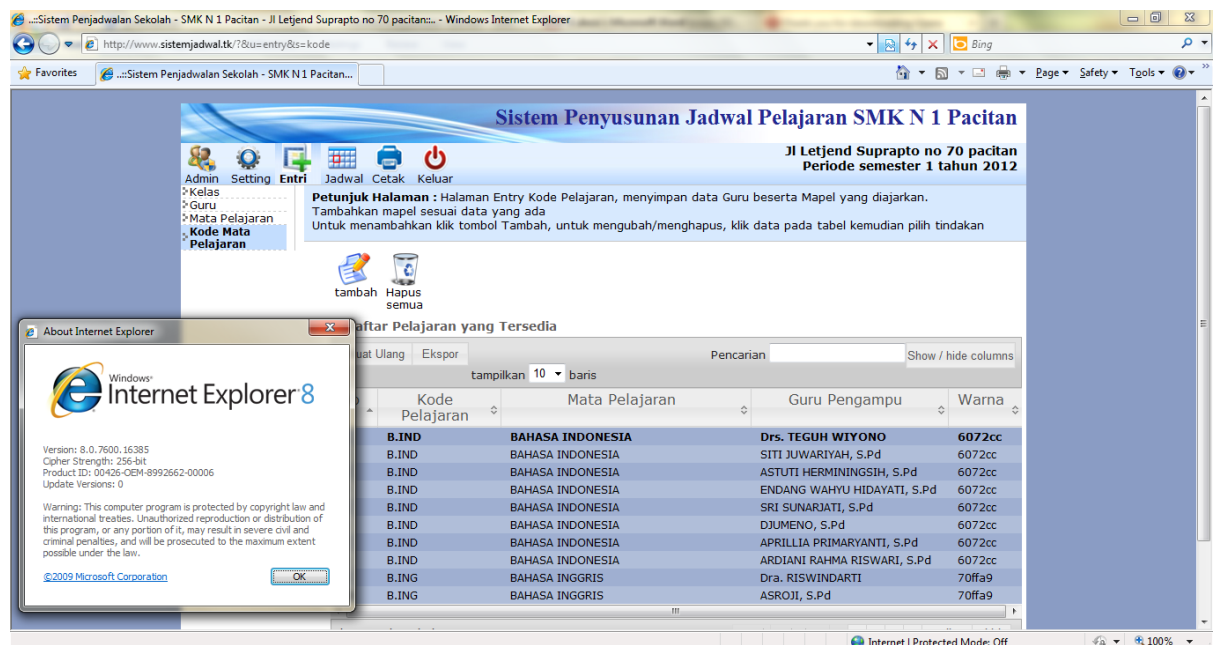
Test case id	Browser compatibility
Purpose	Menguji faktor <i>compatibility</i> dengan <i>web browser Opera</i>
Assumptions	1. Sistem diakses pada web browser Opera
Test data	1. login dengan browser Opera 2. menampilkan data dengan Opera
Steps	1. login sebagai admin /operator 2. klik menu entri → kode pelajaran
Expected result:	1. tampil halaman utama admin 2. tampil halaman data kode pelajaran
Actual result:	1. tampil halaman utama admin 2. tampil halaman data kode pelajaran
Pass/Fail:	Lolos



Gambar 2 : Tampilan pada *Opera*

Tabel 21 : *Test Case Internet Explorer*

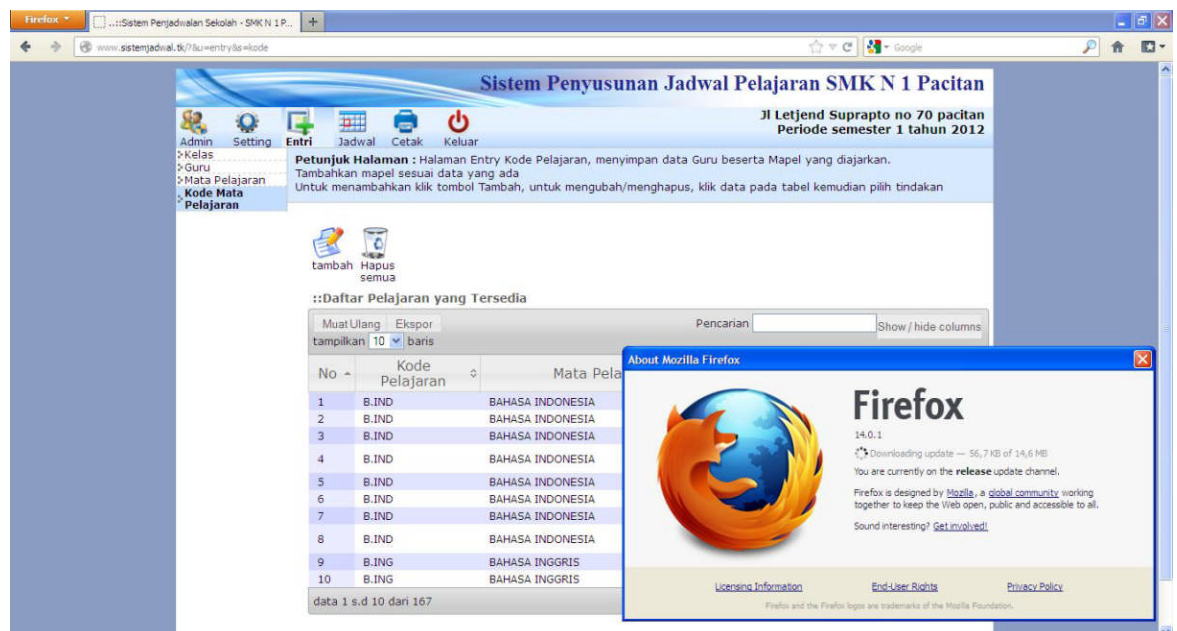
Test case id	Browser compatibility
Purpose	Menguji faktor <i>compatibility</i> dengan <i>web browser Internet Explorer</i>
Assumptions	1. Sistem diakses pada web browser Internet Explorer
Test data	1. login dengan browser Internet Explorer 2. menampilkan data dengan Internet Explorer
Steps	1. login sebagai admin /operator 2. klik menu entri → kode pelajaran
Expected result:	1. tampil halaman utama admin 2. tampil halaman data kode pelajaran
Actual result:	1. tampil halaman utama admin 2. tampil halaman data kode pelajaran
Pass/Fail:	Lolos



Gambar 3 : Tampilan pada *Internet Explorer*

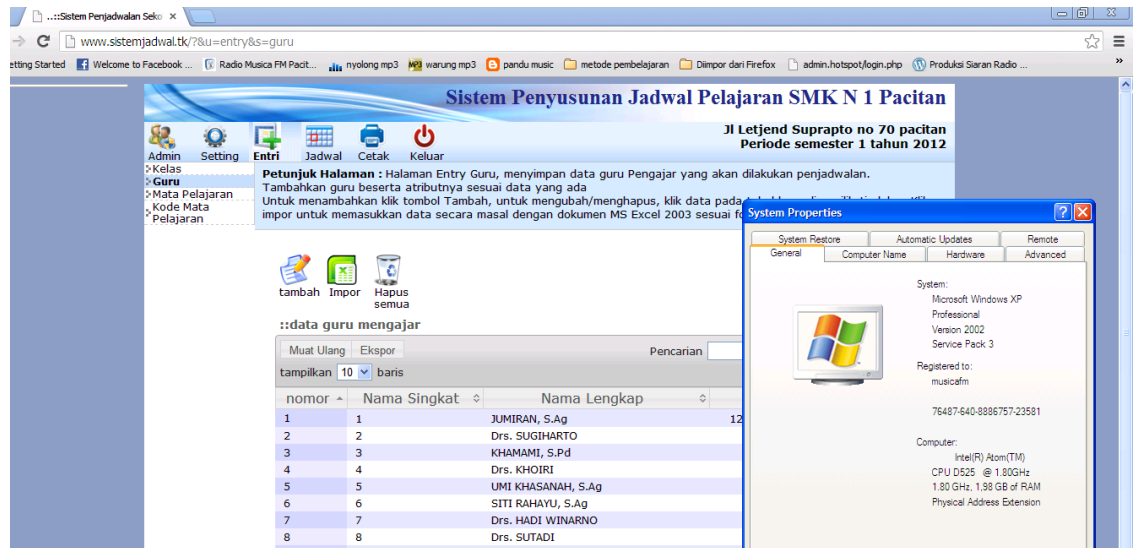
Tabel 22 : *Test Case Mozilla Firefox*

Test case id	Browser compatibility
Purpose	Menguji faktor <i>compatibility</i> dengan <i>web browser Mozilla Firefox</i>
Assumptions	1. Sistem diakses pada web browser <i>Mozilla Firefox</i>
Test data	1. login dengan browser <i>Mozilla Firefox</i> 2. menampilkan data dengan <i>Mozilla Firefox</i>
Steps	1. login sebagai admin /operator 2. klik menu entri → kode pelajaran
Expected result:	1. tampil halaman utama admin 2. tampil halaman data kode pelajaran
Actual result:	1. tampil halaman utama admin 2. tampil halaman data kode pelajaran
Pass/Fail:	Lolos

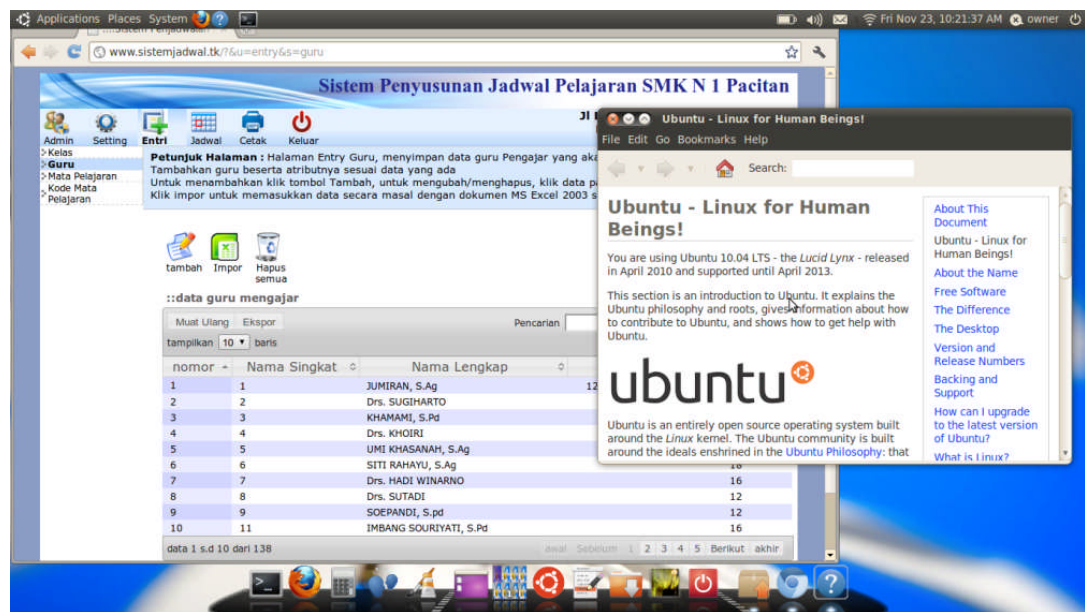


Gambar 4 : Tampilan pada *Mozilla Firefox*

HASIL PENGUJIAN COMPATIBILITY (OPERATING SYSTEM)



Gambar 5 : Tampilan pada *Windows XP*



Gambar 6 : Tampilan pada *Ubuntu Lucid*

SURAT PERNYATAAN VALIDASI KUESIONER DOSEN

REKAPITULASI HASIL UJI COBA KUESIONER

KUESIONER UNTUK AHLI SOFTWARE

SURAT PERNYATAAN VALIDASI PERANGKAT LUNAK

REKAPITULASI DATA HASIL PENELITIAN DESKRIPTIF

CONTOH HASIL CETAK JADWAL PELAJARAN

SURAT IJIN PENELITIAN FAKULTAS TEKNIK

SURAT IJIN PENELITIAN PROVINSI DIY

SURAT IJIN PENELITIAN PROVINSI JAWA TIMUR

SURAT IJIN PENELITIAN DINAS PENDIDIKAN KAB PACITAN

SURAT IJIN PENELITIAN SMK NEGERI 1 PACITAN

**SURAT KETERANGAN TELAH MELAKSANAKAN PENELITIAN DI
SMK NEGERI 1 PACITAN**